

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

СОВРЕМЕННЫЕ ПРОБЛЕМЫ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

**Материалы VI Республиканской научной конференции
молодых ученых и студентов
26 - 28 ноября 2009 года**

Часть II

Брест 2009

УДК 517+519+004
ББК 22.11/.19+32.97

Редакционная коллегия:

В.С. Рубанов, БрГТУ, к.ф.-м.н., доцент – главный редактор
В.А. Головкин, БрГТУ, д.т.н., профессор – зам. главного редактора
Р.Х. Садыков, БГУИР, д.т.н., профессор
Н.В. Лазакович, БГУ, д.ф.-м.н., профессор
А.И. Калинин, БГУ, д.ф.-м.н., профессор
А.А. Дудкин, ОИПИ НАНБ, к.т.н., с.н.с.
В.Ф. Савчук, БрГУ, к.ф.-м.н., доцент
В.М. Ракецкий, БрГТУ, к.ф.-м.н., доцент
С.С. Дереченник, БрГТУ, к.т.н., доцент
Л.П. Махнист, БрГТУ, к.т.н., доцент
С.И. Парфомук, БрГТУ, к.т.н., доцент
Ю.В. Савицкий, БрГТУ, к.т.н., доцент
Д.А. Костюк, БрГТУ, к.т.н., доцент

Рецензенты:

кафедра информатики и прикладной математики БрГУ им. А.С. Пушкина;
зав. кафедрой алгебры и геометрии БрГУ им. А.С. Пушкина Матысик О.В., к.ф.-м.н., доцент.

Современные проблемы математики и вычислительной техники: материалы VI Республиканской научной конференции молодых ученых и студентов, Брест, 26–28 ноября 2009 г. – Брест: БрГТУ, 2009. – Ч. 2, – 188 с.

ISBN 978-985-493-133-3 (Ч. 2)
ISBN 978-985-493-131-9

Представлены тезисы докладов по современным проблемам информационных технологий в научных и прикладных исследованиях, моделирования нелинейных динамических процессов, распознавания образов и анализа изображений, искусственного интеллекта и нейронных сетей, аналитических и численных методов исследований в математике и их приложениях.

Издается в 2-х частях. Часть 2.

УДК 517+519+004
ББК 22.11/.19+32.97

ISBN 978-985-493-133-3

Учреждения образования
© «Брестский государственный технический университет», 2009

СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В НАУЧНЫХ И ПРИКЛАДНЫХ ИССЛЕДОВАНИЯХ

UDC 004.722.25

EXPERIMENTAL CONVERGENCE STUDY OF A GENETIC ALGORITHM USING CONSISTENCY-PRESERVING GENETIC OPERATORS FOR TREE-LIKE CHROMOSOMES

Glebov A., Strykeleu D.
Belarusian State University, Minsk

Genetic algorithms (GAs) constitute a class of robust and efficient optimization techniques capable of solving complex scientific and engineering problems. Key challenges to successful application of GAs in each particular case include the choice of an adequate chromosome representation and the design of the appropriate genetic operators.

A recent study has shown the possibility of employing GAs for network traffic optimization [1–3]. The tree-like chromosome representation seems a natural choice in this case, reflecting the hierarchical character of routing in contemporary networks. However, existing implementations of genetic operators prove to be unsuitable with regard to tree-like structures as they tend to break the consistency of the encoded topologies. Consequently, specialized consistency-preserving operators need to be designed. Two such implementations are presented further.

The proposed crossover operator is based on the idea of the traditional uniform crossover but manipulates matrices, whereas the standard implementation deals with bit strings. The construction of child chromosomes involves the analysis of the already constructed fragments. For each node:

1. ancestor nodes are determined in both parent chromosomes;
2. an attempt is made to connect the current node to one of the nodes found at step 1.

The connection is governed by the following rules:

1. If the current node is not connected to either of the supposed ancestor nodes in both child chromosomes, it will be attached randomly to one of them in one child chromosome and to the other in the second child chromosome;
2. If the current node is connected to one of the supposed ancestor nodes in either child chromosome but not connected to the other ancestor node in the same chromosome, it will be attached to the latter node;
3. If there is already a path from the current node to both its supposed ancestor nodes, the procedure will stop and return the exact copies of the parent chromosomes.

The proposed mutation operator swaps two nodes of the same type within a tree.

In order to study the convergence properties of a GA that uses these operators we ran a series of computer experiments with the test problem being resource allocation for videoconferencing in a LAN of 79 nodes. The population size was set to 140 (the value found in a series of preliminary experiments). The standard crossover rate of 0,5 was chosen according to [4]. Proportionate selection and fitness-based replacement schemes were used for simplicity.

The results (fig. 1, 2) suggest that in a certain region of the parameter space the GA converges quickly (within 100 generations) and stably (reaching the global maximum in 98–100% of all runs).

Fig. 1 – The number of generations required for the GA to converge

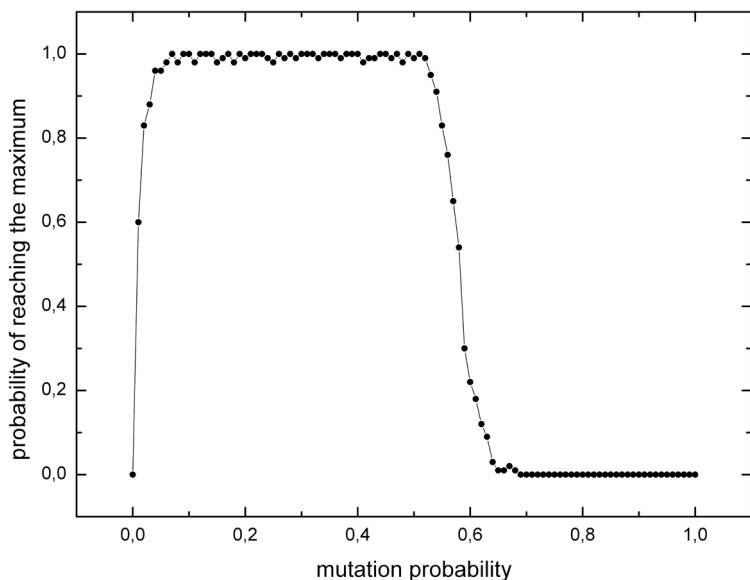
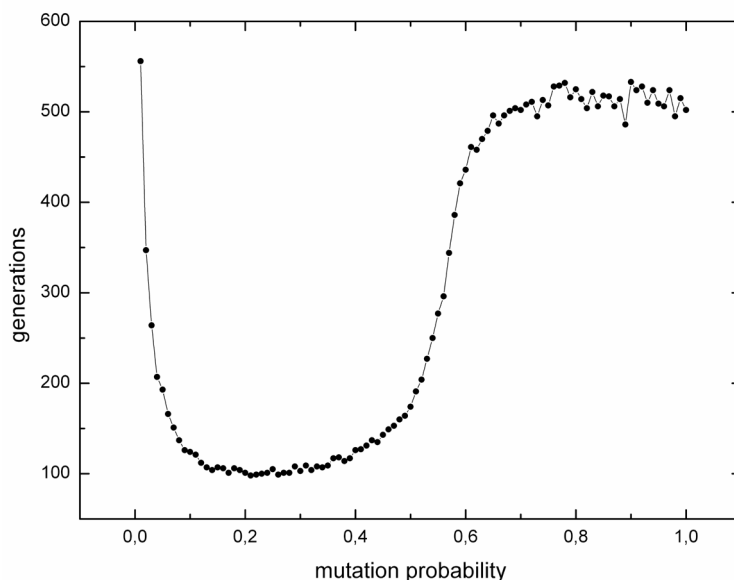
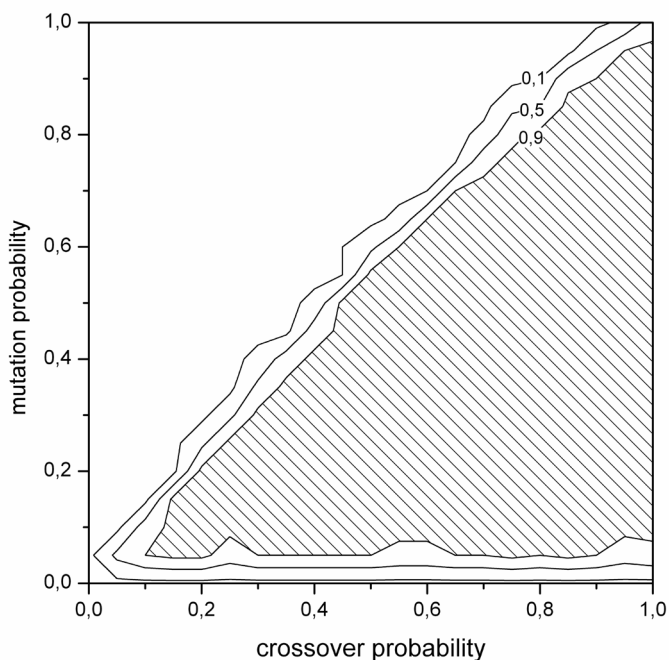


Fig. 2 – The probability of converging to the global optimum

Further experiments were undertaken to determine the convergence region in the crossover-mutation parameter space. The obtained results (fig. 3) allow us to formulate a simple practical rule: a GA that uses the proposed genetic operators converges to the global optimum if the mutation probability is lower than that of crossover, but higher than 10%.

Fig. 3 – The convergence region in the crossover-mutation parameter space. Shaded is the area where the probability of locating the global optimum approaches 100%



Reference

1. Flores, S.D. Telecommunication Network Design with Parallel Multi-objective Evolutionary Algorithms / S.D. Flores, B.B. Cegla, D.B. Cáceres // LANC'03: Proceedings of the 2003 IFIP/ACM Latin America conference on network research, La Paz, Bolivia, October 3–5, 2003. – La Paz, 2003. – P. 1–11.
2. Konak, A. Designing Resilient Networks Using a Hybrid Genetic Algorithm Approach / A. Konak, A.E. Smith // GECCO'05: Proceedings of the 2005 conference on Genetic and evolutionary computation, Washington, USA, 25–29 June 2005. – Washington, 2005. – P. 1279–1285.
3. Y.I. Varatnitsky, D.A. Strykeleu. Network design using linkage learning genetic algorithms // Informatics. – 2006. – № 2. – P. 116-123.
4. Beasley D., Bull D. R., Martin R. R. An overview of genetic algorithms: part 1, fundamentals // University Computing. – 1993. - № 15. – P. 58–69.

УДК 681.01(076.6)

СОВРЕМЕННЫЕ ТЕХНОЛОГИИ WEB-ПРОГРАММИРОВАНИЯ: CMS JOOMLA

Брикет Я.С.

УО «Белорусский государственный экономический университет», г.Минск

В современном мире бизнеса ничто не стоит на месте, любое промедление может привести к поражению в конкурентной среде. И поэтому сегодня действительно актуально и перспективно использование информационных технологий в научных и прикладных разработках. Web-программирование является важным направлением в современной IT-среде, и здесь очень важно найти удобный и эффективный способ создания сайтов. Так как реализация многих проектов трудоемка и очень затратна, то целесообразно использование системы управления содержимым «Joomla», написанной на языках PHP и JavaScript, использующей в качестве хранилища содержания базу данных MySQL. Joomla является свободным программным обеспечением, защищённым лицензией.

Название «Joomla!» фонетически идентично слову «Jumla», которое в переводе с языка суахили означает «все вместе» или «единое целое», что отражает подход разработчиков и сообщества к развитию системы. Joomla! CMS (система управления содержимым) - одна из наиболее мощных систем управления сайтом среди существующих на нашей планете.

CMS Joomla! включает в себя различные инструменты для изготовления web-сайта. Важной особенностью системы является минимальный набор инструментов при начальной установке, который обогащается по мере необходимости. Это снижает загромождение административной панели ненужными элементами, а также снижает нагрузку на сервер и экономит место на хостинге.

На движке Joomla можно создавать:

- Личные и семейные домашние страницы
- Сайты малого бизнеса
- Корпоративные сайты и порталы
- Социальные порталы
- Интернет-магазины
- Журналы и газеты
- Форумы

Характеристики Joomla!:

- Полностью основанный на БД движок с использованием PHP/MySQL.
- Секции новостей, продукции или услуг легко редактируемы и управляемы.

- Разделы могут быть добавлены авторами.
- Полностью настраиваемые схемы расположения элементов, включая левый, правый и центральный блоки меню.

- Форум/Опросы/Голосования для эффективной обратной связи.
- Работа под Linux, FreeBSD, MacOSX, Solaris, AIX, SCO, WinNT, Win2K.

Возможности администрирования:

- Возможность создавать неограниченное количество страниц.
- Четкая организация структуры сайта.
- Для каждой динамической страницы можно создать свое описание и ключевые слова в целях повышения рейтинга в поисковых системах.
 - Начало и окончание публикации любых материалов можно запрограммировать по календарю.
 - Возможность ограничить доступ к определенным разделам сайта только для зарегистрированных пользователей.
 - Секции новостей, продукции или услуг легко редактируемы и управляемы.
 - Полностью настраиваемые схемы расположения элементов по 5 областям, что позволяет сделать оригинальный дизайн.
 - Закачка изображений при помощи браузера в вашу собственную библиотеку - для последующего использования с любого места сайта.
 - Опросы и голосования для эффективной обратной связи.
 - Различные модули - такие как последние новости, счетчик посещений, подробная статистика посещений, гостевая книга, форум и т.д.
 - Причем Вы сами выбираете, будут ли показаны эти модули и на каких страницах.
 - Возможность создания не одной, а нескольких форм обратной связи для каждого контакта.
 - Изменение порядка объектов, включая новости, вопросы, статьи и т.д.
 - Генератор показа случайной новости.
 - Модуль приёма от удалённых авторов новостей, статей и ссылок.
 - Иерархия объектов - количество секций, разделов, подразделов и страниц, зависит от вашего желания.
 - Библиотека изображений позволит Вам хранить все Ваши GIF- и JPEG-файлы под рукой для лёгкого доступа.
 - Менеджер рассылки новостей. Выбирайте из более чем 360 служб рассылки новостей по всему миру.
 - Менеджер архива. Помещайте Ваши старые статьи в архив, чем просто удалять их.
 - Возможность распечатать или отправить другу на e-mail любую статью с сайта.
 - Выбор из 3 визуальных редакторов, что упрощает редактирование материалов до уровня редактирования текста в программе Word.
 - Предварительный просмотр перед окончательным размещением.
 - Возможность легкой смены дизайна.
 - Экономное использование места на сервере за счет использование базы данных MYSQL.
 - Возможность использования адресов страниц, адаптированных для лучшей индексации всеми поисковыми системами.
 - Система подходит как для небольшого, так и для крупного корпоративного сайта или информационного проекта.

Таким образом, установив эту систему на свой веб-сайт, Вы получаете полный контроль над обновлением информации и развитием Вашего веб-сайта. И для этого совсем не нужно иметь никаких специальных знаний.

Имея огромное количество расширений, Joomla! позволяет удовлетворить требования заказчика в любых аспектах и разработать сайт с учетом удобного представления информации, расположения основных модулей и администрирования сайтом.

Преимущества CMS Joomla

Система управления сайтом Joomla - одна из наиболее мощных систем управления сайтом среди существующих на нашей планете. CMS Joomla проверена на надежность и безопасность, о чём свидетельствуют соответствующие международные сертификаты. В 2007 году в конкурсе "Open Source Content Management System Award" CMS Joomla заняла 1-е место. Сейчас появилось очень много движков для сайтов, все они разные по качеству и функциональности. И зачастую перед Вами встает задача – выбрать движок для сайта путем экспериментов или заказать создание сайта. В первом варианте Вы потратите уйму времени, а во втором – средств. Благодаря Joomla Вы абсолютно свободны в своих творческих решениях и способны воплотить любой замысел еще легче, чем ранее.

Литература

1. Колисниченко Д.Н. Joomla! 1.5. Руководство пользователя / Д.Н. Колисниченко – М.: [Диалектика](#), 2009. – С. 224.
2. Хаген Граф Создание веб-сайтов с помощью Joomla! 1.5 = Building Websites with Joomla! 1.5. – М.: «Вильямс», 2009. – С. 304.
3. Бэрри Норт Joomla! Практическое руководство = Building a Successful Joomla! Powered Website. – М.: «Символ-Плюс», 2008. – С. 448.
4. Дэн Рамел Joomla! Самоучитель = Beginning Joomla! From Novice to Professional. — Санкт-Петербург: «БХВ-Петербург», 2008. – С. 448.
5. <http://joomlaportal.ru/>

УДК 519.171.1

КОМПЬЮТЕРНАЯ ГЕНЕРАЦИЯ ТРЕХСВЯЗНЫХ ОДНОРОДНЫХ ПЛОСКИХ ГРАФОВ, НЕ ИМЕЮЩИХ ГАМИЛЬТОНОВА КОНТУРА

Брич А.Л., Мухоровский С.В.

УО «Брестский государственный технический университет», г. Брест

В монографии Ф. Харори [1] дается указание на работу Тейта, который высказал предположение, что каждый трехсвязный плоский граф содержит остовный простой цикл. Это сразу же означало бы справедливость гипотезы четырех красок. В дальнейшем Татт [2] показал, что это неверно, приведя пример трехсвязного плоского графа с 46 вершинами, не являющимся гамильтоновым. В [2] приведен однородный кубический трехсвязный плоский граф с числом вершин 42, который также не имеет гамильтонова контура. В монографии Грюнбаума [3] приводится наименьший известный в настоящее время негамильтонов трехсвязный плоский граф, имеющий 38 вершин.

Следует предположить, что среди трехсвязных однородных плоских графов степени 3 довольно много негамильтоновых. До настоящего времени попытки построить подобные графы являлись «штучными», то есть «ручным» творчеством отдельных исследователей.

Целью настоящей работы является автоматизация построения названных графов и получения графа с числом вершин меньше 38. Для этого разрабатывается комплекс из четырех последовательно работающих программ:

-программа 1 генерирования гамильтоновых простых однородных графов степени 3 с заданным четным числом вершин;

-программа 2 выделения только трехсвязных графов из массива полученных программой 1 графов;

-программа 3, которая в гамильтонов трехсвязный простой граф, полученный на этапе 2, вносит две новые вершины: одна из которых лежит на гамильтоновом контуре, а вторая находится вне его.

Обоснованием введения программы 3 является следующее предположение:

Предположение 1. Все однородные графы с числом вершин n и степенью 3, не имеющие мостов, имеют гамильтонов контур либо контур максимальной длины из $n-1$ вершин.

Примером может служить граф Петерсена, либо упомянутые выше три однородных графа степени 3.

Все полученные на этапе 3 графы проверяются программой 4 на наличие гамильтонового контура. В результате графы, не имеющие гамильтонового контура, составляют искомое множество.

Данный комплекс программ позволит вычислить долю трехсвязных плоских графов в общем числе плоских однородных графов степени 3, а также установить долю таких графов в общем числе однородных кубических графов. Отметим, что последнее возможно выполнить только для однородных кубических графов с $n < 24$, т.к. число таких графов велико и последнее рассчитанное значение выполнено для $n = 24$ [4]. Согласно [4] время расчета составило около 10 часов.

В данном проекте также реализуется 2 вида генерации графов: путём послойного наращивания количества вершин в плоских трёхсвязных графах, а также путём склейки плоских трёхсвязных графов (K_4^3 , K_6^3 , K_8^3 ,...) и различных перестановок. В ходе исследований планируется определение возможности генерации новых, ранее неизвестных плоских трёхсвязных графов, а также более эффективного способа их генерации, который способен охватить большее множество генерируемых графов и за меньшее время работы.

Литература

1. Харари, Ф. Теория графов / Ф. Харари –М.: Мир, 1973. – С. 87.
2. Емиличев, В.А. Лекции по теории графов / В.А. Емиличев, О.И. Мельников, В.И. Сорванов, Р.И. Тышкевич – Москва: Наука, 1990. – С. 203.
3. Grunbaum, B. Contex polytopes, Wiley, New York, 1967. – P. 359.
4. Meringer, M. Erzeugung regularer graphen. Bayreuth, 1996. – S. 36.

УДК 681.3

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ

Брич А.Л.

УО «Брестский государственный технический университет», г. Брест

Обучение программированию предполагает выработку у обучаемых специальных навыков работы с компьютером, включая разработку алгоритмов решаемых задач, кодирование их на языках программирования. Для обучения разработке алгоритмов, написанию и отладке программ наряду с обучающими системами в настоящее время преимущественно используются существующие системы программирования.

При этом обучение алгоритмизации, тесно связанное с оценкой корректности алгоритмов и поиском ошибок, как правило, производится вручную. Качество обучения алгоритмизации, его эффективность может быть в значительной мере повышена использованием компьютерных сред, обеспечивающих исполнимость алгоритмов по аналогии с системами программирования.

Важная часть такой среды - входной язык, лингвистические средства. Так текст разрабатываемого алгоритма должен быть инвариантен к языкам программирования высокого уровня, читаемым, легко контролируемым вручную и автоматически в процессе разработки и в итоге должен быть готовым документом для кодирования на подходящем языке программирования.

В работе, которая является продолжением и развитием [1, 2], приводятся результаты апробации принципов построения входного языка и соответствующих инструментальных средств. Лингвистическое обеспечение такой системы должно быть построено на принципах структурного программирования. Кроме этого, получаемый текст должен обладать исполнимостью. Входной язык, соответствуя базовым принципам построения языков программирования, должен основываться на минимальном наборе изобразительных средств, управляющих структур, отображающих соответствующие математические понятия и правила пользования ими.

Используется процедурный механизм построения алгоритмов

`<программа> ::= <основной_блок> <подпрограммы> ,`
`<подпрограмма> ::= <функция> | <процедура> .`

Для повышения читаемости разрабатываемых текстов, в частности, должны использоваться: - комментарии, "встраиваемые" в текст без выделения специальными служебными символами, наряду с традиционными строчными комментариями; - гибкая и удобная в использовании система образования идентификаторов с применением "встраиваемых" комментариев; - "подсветка синтаксиса" текста алгоритма при работе в редакторе для повышения читаемости текста, его восприятия.

Указанные "встраиваемые" комментарии должны записываться прописными буквами кириллицы, что позволит склонять используемые в тексте алгоритма идентификаторы в зависимости от контекста, и соответственно формулировать осмысленные тексты (совокупности команд) по правилам, близким к правилам русского языка. Соответственно

`<идентификатор> ::= [<приставка>] <корень> [<окончание>] { ["_" [<приставка>]`
`<корень> [<окончание>] } .`

Здесь

`<корень> ::= <прописная_буква> { [<прописная_буква> | "_" | <цифра>] } ,`

`<приставка> ::= <встраиваемый_комментарий> ,`

`<окончание> ::= <встраиваемый_комментарий> ,`

`<встраиваемый_комментарий> ::= <строчная_буква> { [<строчная_буква >] } .`

Дополнительно в язык должны быть введены средства описания глобальных и локальных объектов, в том числе, байтового, символьного, логического и файлового типов. Указанный перечень является достаточным для разработки и описания класса типовых алгоритмов.

Набор операторов языка должен отличаться единым стилем описания и должен содержать операторы следующих типов: `<оператор_присваивания>`, `<оператор_вызова>`, `<условный_оператор>`, `<оператор_цикла_с_предусловием>`, `<оператор_цикла_с_постусловием>`, `<оператор_цикла_с_параметром>`. При редактировании алгоритма должна обеспечиваться работа с блоками текста, буфером обмена. Соответственно инструментальные средства должны обеспечивать: редактирование алгоритмов; исполнение алгоритмов, включая контроль корректности текстов алгоритмов, трансляцию и получение исполнимого кода, управление его выполнением; пошаговое выполнение алгоритмов в целях обучения и отладки; генерацию программ на языках высокого уровня.

Работа системы макетировалась с использованием языков высокого уровня (Pascal, Delphi). Система Windows-ориентирована. Среда пользователя - интегрированная, многооконная, русифицированная. Получаемый в результате разработки текст алгоритма позволяет автоматически генерировать код (функционально-адекватное внутреннее представление алгоритма), исполняемый, например, внутренней виртуальной машиной, которая и реализует интерфейс взаимодействия пользователя с алгоритмом в процессе его разработки и отладки.

Литература

1. Муравьев, Г.Л. Входной язык автоматизированной системы обучения (АСО) алгоритмизации // Новые информационные технологии в образовании: материалы 3 международной конференции NITE' 98, Минск, 1998 г.: в 3 т. / БГЭУ. – Минск, 1998. - Т.2. - С. 83–88.

2. Муравьев, Г.Л. Автоматизация обучения алгоритмизации и программированию / Г.Л. Муравьев, С.В. Мухов // Вести ИСЗ. - 2000. – № 3. - С. 24-29.

УДК 004.6

ОБ ОРГАНИЗАЦИИ СИСТЕМЫ ТЕСТИРОВАНИЯ

Ванюков С.В.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Силаев Н.В., доцент*

Начиная разработку системы тестирования TENMA, была поставлена задача создать, по возможности, универсальный инструмент, который можно будет легко расширить и приспособить для любой системы тестирования. В качестве инструмента разработки был выбран язык C# на платформе Visual Studio .NET 2.0. Предоставляемые ею средства для сетевого программирования, построения классов и шифрования передаваемых данных оказались подходящими для решения поставленной перед нами задачи.

Основное внимание при проектировании уделяется открытости системы и простоте её обновления и инсталляции. Разрабатываемая система позволит: строить древовидно организованные структуры связей, моделирующие диалог естественного общения при опросе; обеспечить поддержку импорта тестов формата TQF, который применяется в настоящее время в системе теоретического тестирования на математическом факультете БрГУ.

Пакет состоит из двух частей — клиентской и серверной. Клиентская часть инсталлируется на все компьютеры, за которыми производят тестирование. Серверная часть устанавливается только на один – компьютер администрирования.

Общие требования, выдвигаемые нами к системе, таковы:

Удобство администрирования.

Совместимость. Система должна поддерживать уже существующие форматы тестов или поставляться с конвертером.

Полнота. Всё необходимое для работы системы должно быть включено в её дистрибутив, чтобы до предела упростить процесс инсталляции.

Универсальность. Система должна поддерживать различные типы вопросов и ответов. Например, вопрос может содержать иллюстрации и текстовое форматирование, ответ должен быть только одним из предложенных, набором из нескольких предложенных вариантов или вводиться с клавиатуры.

Лёгкость тестирования.

Защита данных. Обе составляющие системы (клиентская и серверная) должны быть устойчивы к «атакам» различного рода.

Мобильность.

Отказоустойчивость. Система должна чётко и грамотно реагировать на сбои в работе сети. В случае ошибки и администратор, и пользователи должны чётко знать возникшую проблему.

УДК 004.891.2

ПРОГРАММА XPERTRULE KNOWLEDGE BUILDER КАК ИНСТРУМЕНТ ПОСТРОЕНИЯ ДЕРЕВЬЕВ РЕШЕНИЙ

Геращенко Е.В.

УО «Белорусский государственный экономический университет», г. Минск

Стремительное развитие информационных технологий, в частности прогресс в методах сбора, хранения и обработки данных, позволил многим организациям собирать огромные массивы данных, которые необходимо анализировать. Объемы этих данных настолько велики, что возможностей экспертов уже не хватает, что породило спрос на методы автоматического исследования (анализа) данных.

Деревья решений – один из таких методов. Первые идеи создания деревьев решений восходят к работам Ховленда (Hoveland) и Ханта (Hunt) конца 50-х годов XX века. Однако основополагающей работой, давшей импульс для развития этого направления, явилась книга Ханта (Hunt, E.V.), Мэрина (Marin J.) и Стоуна (Stone, P.J) "Experiments in Induction", увидевшая свет в 1966г.

Деревья решений – это способ представления правил в иерархической, последовательной структуре, где каждому объекту соответствует единственный узел, дающий решение.

Область применения «деревья решений» в настоящее время широка, но все задачи, решаемые этим аппаратом, могут быть объединены в следующие три класса:

- Описание данных: Деревья решений позволяют хранить информацию о данных в компактной форме, вместо них мы можем хранить дерево решений, которое содержит точное описание объектов.
- Классификация: Деревья решений отлично справляются с задачами классификации, т.е. отнесения объектов к одному из заранее известных классов. Целевая переменная должна иметь дискретные значения.
- Регрессия: Если целевая переменная имеет непрерывные значения, деревья решений позволяют установить зависимость целевой переменной от независимых(входных) переменных. Например, к этому классу относятся задачи численного прогнозирования(предсказания значений целевой переменной).

На сегодняшний день существует значительное число алгоритмов, реализующих деревья решений CART, C4.5, NewId, ITrule, CHAID, CN2 и т.д.

Деревья решений являются действенным инструментом интеллектуального анализа данных и успешно применяются для решения практических задач во многих областях, таких как банковское дело, промышленность, медицина и т.д.

С помощью деревьев решений можно реализовывать ряд экономических задач, а также строить системы поддержки принятия решений менеджеров различного уровня. Существует множество программ для проектирования деревьев решений.

Одной из таких программ является XpertRule Knowledge Builder. Она представляет собой мощную среду для разработки и использования основанных на знаниях приложений. Основанные на знаниях приложения – это компоненты программного обеспечения, которые включают различные правила, накопленный опыт, ноу-хау, методики, стандарты и положения.

Всё вышеперечисленное представляет собой «бизнес-правила». Автоматизация бизнес-правил, в свою очередь, – это средство, с помощью которого развивается «экономика знаний» и «электронный бизнес». В программе Knowledge Builder интегрированы различные технологии, имеется обширная среда разработки и возможности разностороннего её использования.

На примере экономической задачи рассматриваются возможности программы, а также особенности построения деревьев решений, как систем поддержки принятия решений.

Создаётся экспертная система «Awarding bonuses» для решения о премировании или депремировании сотрудников компании. Решение принимается на основе следующих факторов: условия работы (condition of work), график работы (operating mode), перевыполнение плана (overfulfillment of plan), количество отработанных выходных и праздничных дней (amount) и некоторые другие дополнительные условия (additional parameters).

На начальном этапе вводятся сами факторы, затем их значения.

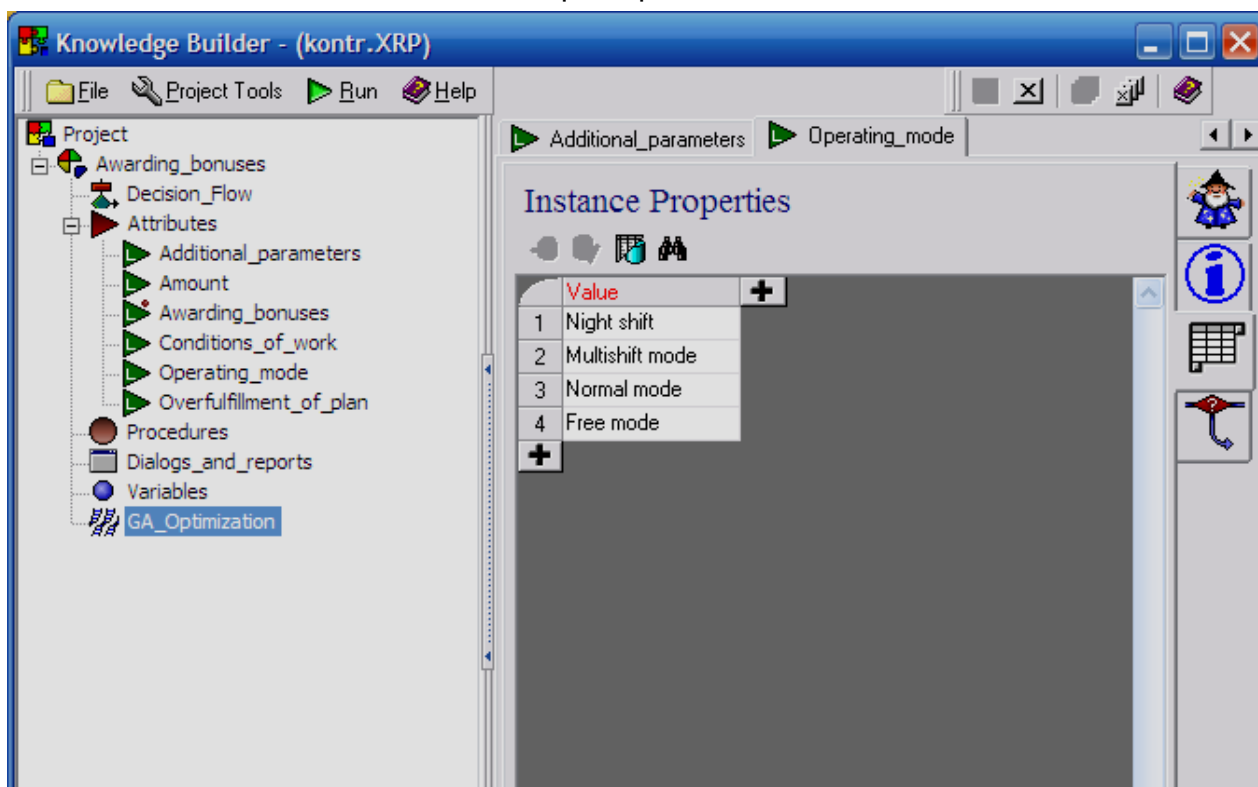


Рисунок 1 – Факторы, на основании которых строится система

Затем строится само дерево решений на основании так называемых «бизнес-правил».

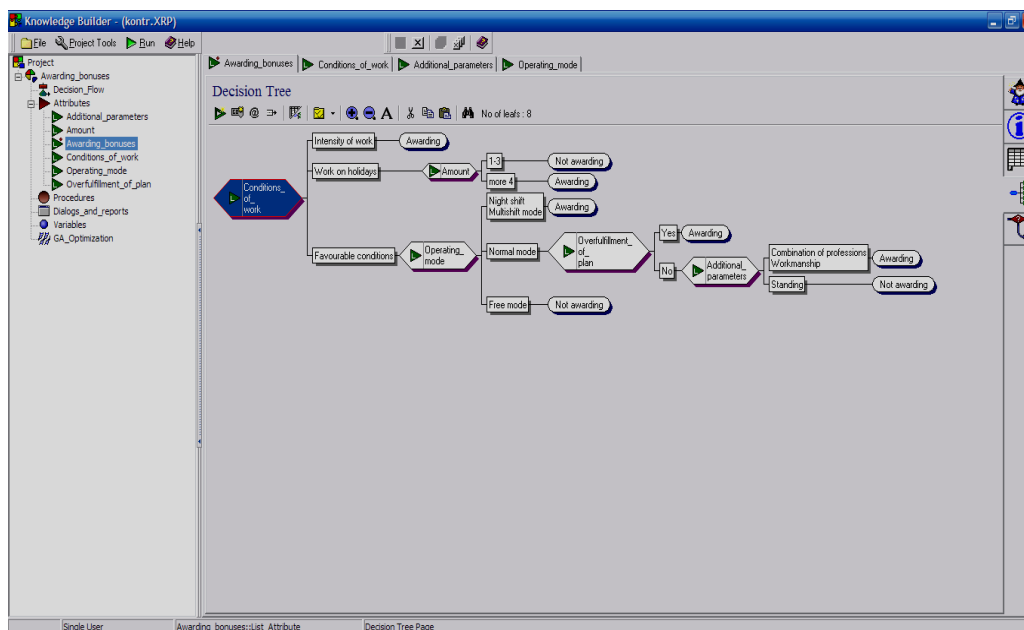


Рисунок 2 – Дерево решений

В результате чего получаем экспертную систему. При помощи этой системы можно быстро определить, начислять или не начислять премию тому или иному работнику при определённых условиях.

Экспертная система имеет следующий вид.

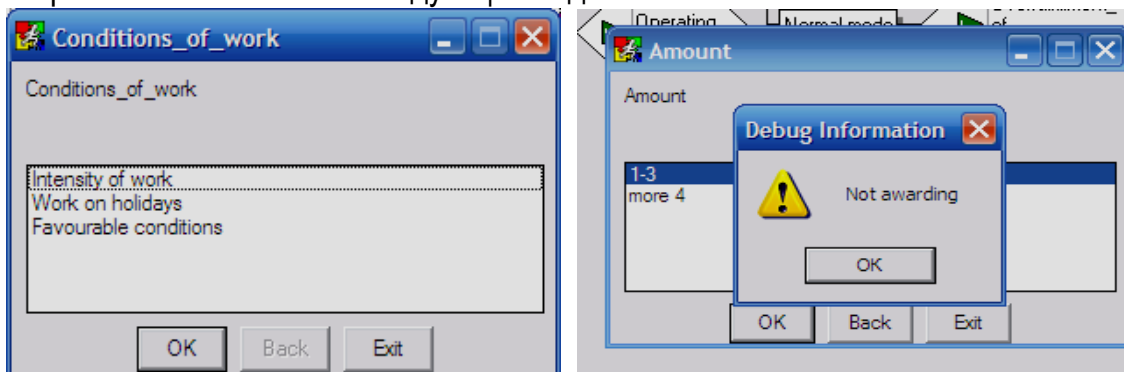


Рисунок 3 – Система поддержки принятия решения о премировании работников

Полученная система будет полезна работникам отдела кадров, бухгалтерии, а также менеджерам по управлению персоналом. Безусловно, она помогает увеличить скорость и затратить меньше времени для принятия решения.

Как известно, в последнее время основной упор делается не только на качественное, но и быстрое выполнение поставленных задач. Чтобы этого достичь, необходимо автоматизировать рутинную, повторяющуюся работу. Тогда менеджер любого уровня будет больше времени затрачивать на решение более важных и глобальных задач.

Программа XpertRule Knowledge Builder является хорошим помощником и качественным инструментом создания подобных систем. Реализации задач различного плана существенно упростит и ускорит работу многих процессов управления и сделает экономику на шаг ближе к информационной.

Литература

1. Синявская О.А., Железко Б.А. Изучение технологии создания экспертных систем с помощью оболочки «XpertRule Knowledge Builder» // Теория та методика навчання математики, фізики, інформатики: Збірник наукових праць. Випуск V: В 3-х томах.— Кривий Ріг: Видавничий відділ НМетАУ, 2005.—Т. 3: Теорія та методика навчання інформатики.— С. 234 – 239.

УДК: 37:004

ПРИМЕНЕНИЕ СИСТЕМЫ КОМПЬЮТЕРНОЙ АЛГЕБРЫ MATHEMATICA НА УРОКАХ МАТЕМАТИКИ

Герман Ю.В.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

Многие учебные учреждения мира используют систему компьютерной алгебры Mathematica в процессе обучения. Указанная система является одним из ведущих программных продуктов для обработки числовых, символьных и графических данных; в то же время ее можно применять и в школьном курсе математики. Например, в процессе изучения тем «Многочлены», «Степени и корни», «Уравнения с одной и несколькими переменными», «Неравенства», «Прямоугольная система координат на плоскости», «Числовая последовательность», «Арифметическая, геометрическая прогрессии», «Числовая функция», «Графики функций», «Тригонометрические функции», «Показательная и логарифмическая функции», «Производная», «Первообразная и интеграл» удобно использовать средства системы компьютерной алгебры Mathematica.

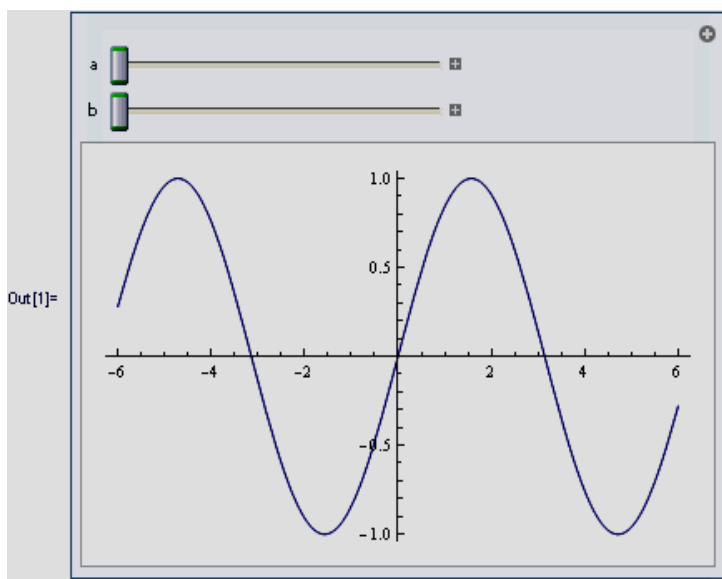
С помощью системы Mathematica можно строить двумерные и трехмерные графики функций. В пакете предусмотрена возможность изображения геометрических объектов, используя стандартные фигуры (примитивы): многоугольники, окружности, их дуги и так далее. Примитивами двумерной графики называют дополнительные указания, вводимые в функцию Graphics для построения некоторых заданных геометрических фигур [1]. Функция Graphics задается в виде: Graphics [primitives, options] — представляет двумерное графическое изображение. Применение примитивов в составе функции Graphics избавит учителя от задания математических выражений, описывающих эти фигуры. Используются следующие примитивы двумерной графики:

Circle[{x, y}, r] — строит окружность с радиусом r и центром в точке {x, y}; Circle[{x, y}, r, {theta1, theta2}] — представляет дугу окружности радиуса r с центром {x, y} и углами концевых точек theta1 и theta2; Disk[{x, y}, r] строит закрашенный круг радиусом r с центром в точке {x, y}; Disk[{x, y}, r, {theta1, theta2}] — строит сегмент круга радиуса r с центром {x, y} и углами концевых точек theta1 и theta2.

Для построения графиков трехмерных поверхностей используется основная графическая функция Plot3D. Mathematica предоставляет большое количество возможностей для отображения результатов в виде диаграмм и графиков. И это является лишь малой частью широких графических возможностей системы.

Еще одним достоинством является справка. Удобная и быстрая навигация поможет быстро найти необходимую функцию, посмотреть примеры [1]. В новой версии Mathematica (6.0 и выше) по некоторым функциям приведено около 50 примеров. Учитель математики может с успехом использовать эти примеры при проведении уроков. Во многих из них не сложно будет разобраться, потому что модуль Manipulate, применяемый в примерах, позволяет манипулировать параметрами, входящими в исходное выражение, не прибегая к редактированию исходного кода. Изменить параметр можно на демонстрируемом изображении, переместив слайдер по полосе в сторону увеличения либо в сторону уменьшения параметра. Эта функция значительно упрощает использование готовых программных продуктов, делая процесс изменения данных быстрым и понятным. Например, при изучении функции $y = \sin(x)$ в школьном курсе математики для демонстрации графика функции учитель может использовать следующую команду Manipulate[Plot[Sin[a x + b], {x, -6, 6}], {a, 1, 4}, {b, 0, 10}].

Функцией Plot строится график функции $y=\text{Sin}(ax+b)$. Параметрами являются a и b .



После функции в фигурных скобках записываются параметры, их начальное и конечное значение, а также может указываться шаг изменения параметра. После того, как записана функция, можно нажать сочетание клавиш «shift+enter». После этого появится результат работы данной команды.

Рисунок 1

На изображении (Рисунок 1) можно видеть две полоски с бегунками, отвечающие за изменение параметров a и b . При перетаскивании бегунков будет видно, как ведет себя график функции $y=\text{Sin}(ax+b)$ (в зависимости от изменения параметров a и b). Применение команды Manipulate к функции Plot3D показывает превосходство пакета по отношению к другим системам компьютерной алгебры, так как в данном пакете можно достаточно просто сделать трехмерную поверхность динамичной.

Использование динамической интерактивности в системе Mathematica позволяет осуществить возможность полного управления графическими объектами. Благодаря этим преимуществам системы, ее применение в образовательном процессе стало еще более доступным. Опираясь на возможности динамизации, можно сделать занятия по математике эффективными, насыщенными, наглядными и интересными. Для реализации более сложных задач учитель может воспользоваться уже готовыми примерами [1].

Например, для подготовки к уроку по теме «числовые последовательности» можно воспользоваться кодом программы, реализация которого поможет учителю в составлении ряда прогрессии и нахождении разности (Рисунок 2). Учитель может задать количество членов числовой последовательности или попросить учащихся назвать следующий член, а затем вывести его на экран, нажав соответствующую кнопку, вывести на экран новую числовую последовательность.

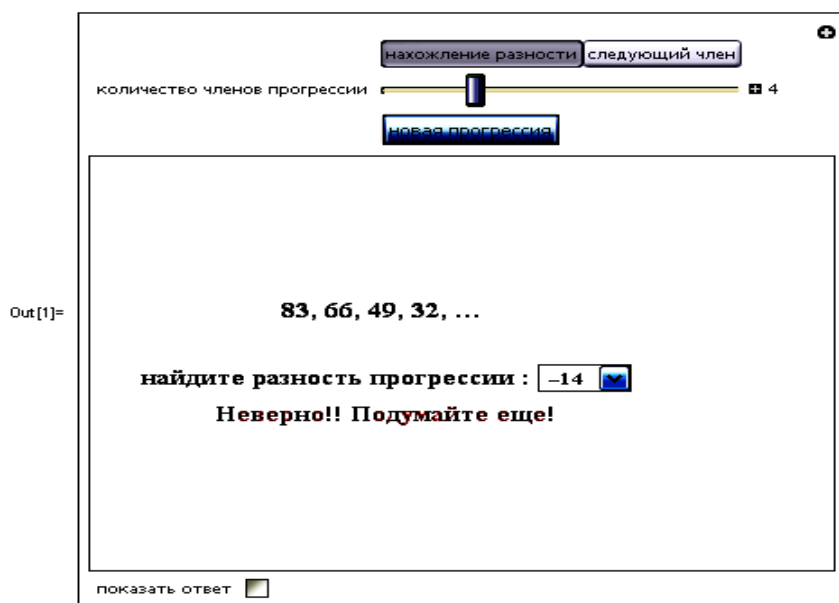


Рисунок 2

Востребованность таких систем, как Mathematica, в современном математическом образовании не вызывает сомнений. Применение системы Mathematica существенно сокращает время решения сложных математических задач. Поэтому её необходимо использовать в школе на факультативных занятиях, уроках, при подготовке к централизованному тестированию, для обсуждения общих подходов и идей решения задач в целях экономии времени. В ходе проводимого исследования «Методика обучения математике учащихся средней школы средствами информационных технологий» нами изучается возможность применения системы компьютерной алгебры Mathematica в школе.

Литература

1. <http://reference.wolfram.com/mathematica/guide/NotebooksAndDocumentsOverview.html>.

УДК 621.9.08

ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС АВТОМАТИЗАЦИИ АКУСТИЧЕСКОГО СПЕКТРАЛЬНОГО АНАЛИЗА

Грисевич Л.Н.

УО «Брестский государственный технический университет», г. Брест

Изучение особенностей отражения акустической волны от среды, характеризующейся заметной дисперсией скорости и сильным поглощением (диссипацией) звука, позволяет неинвазивно и в реальном масштабе времени получать информацию о вязкости среды (или о внутреннем трении для твердых веществ) и ряде производных параметров, а также, опосредованно, о воздействии внешних факторов (температуры, давления, поля). Кроме вискозиметрии, эффект зависимости коэффициента отражения ультразвука от параметров среды может быть использован при измерении влажности почв, грунтов и сыпучих материалов. Также методы и средства акустики диссипативно-дисперсионных сред (ДДС) применимы к решению задач твердотельной электроники, в частности микроэлектроники, к исследованию веществ в области их фазовых переходов, так как там всегда имеет место значительное возрастание поглощения акустических колебаний [1].

В процессе исследований акустики ДДС возникла необходимость создания акустической измерительной установки с автоматизированным вводом данных в персональный компьютер и последующей их обработкой.

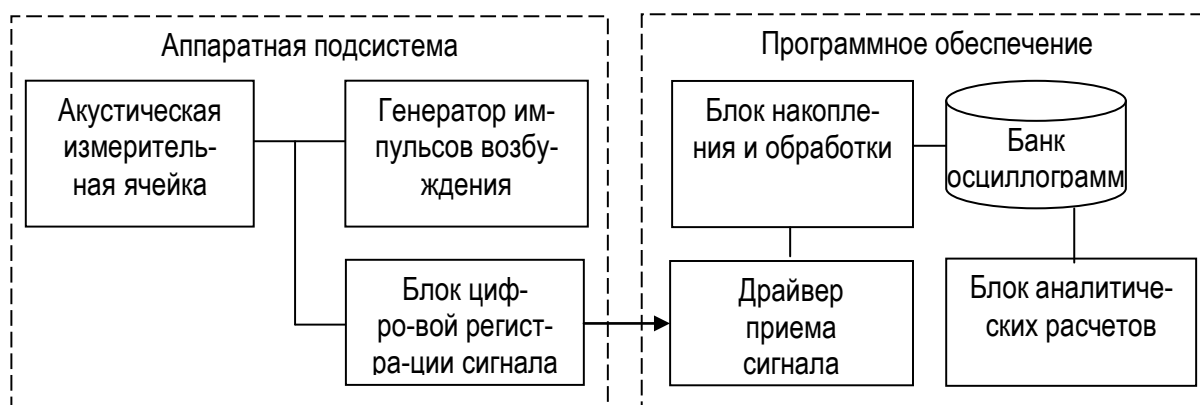


Рисунок 1 — Структура автоматизированной установки

Разработанная на этих принципах стационарная установка изображена на рис. 1. Аппаратная часть установки включает в себя генератор прямоугольных электрических импульсов, питающих акустический тракт подсистемы, блок цифровой регистрации сигнала, представляющий из себя осциллограф с каналом передачи данных в ПК, либо встроенный в компьютер осциллографический адаптер (в проводимых экспериментах использована продукция фирмы Augis), и собственно персональный компьютер с разработанным программным обеспечением.

В состав акустической измерительной ячейки [2, 3] входят ультразвуковой пьезопреобразователь, возбуждаемый поступающими на вход ячейки импульсами и работающий в совмещенном режиме излучения-приема.

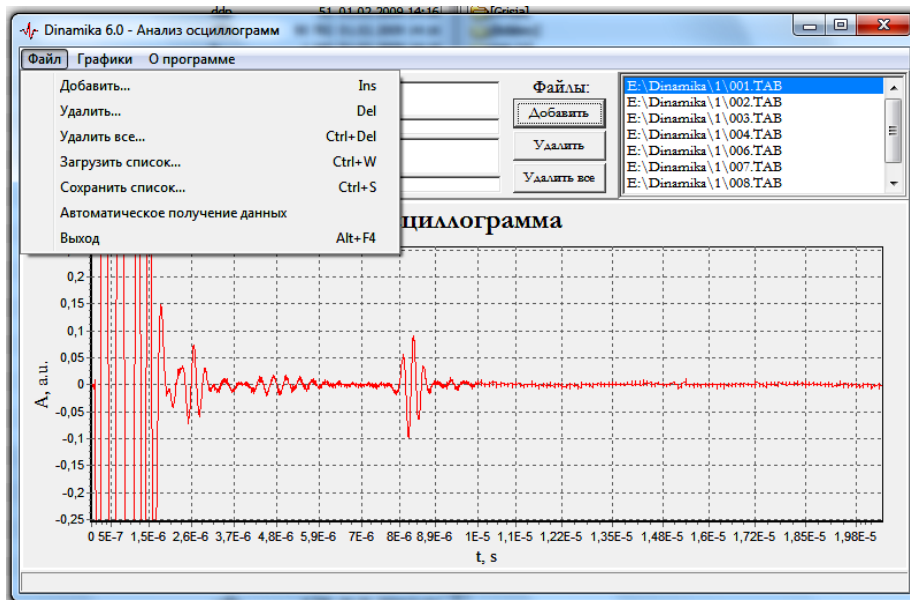
Опционально измерительная ячейка может быть выполнена в виде автономного диагностического модуля с первоначальным накоплением оцифрованных результатов измерений. В его состав входят ультразвуковой пьезопреобразователь, усилитель/ограничитель сигнала, аналогово-цифровой преобразователь, память, передатчик и микроконтроллерный блок управления/индикации. Основной функцией такого устройства является снятие первичных показателей измерений и сохранение их в энергонезависимую память с последующей передачей в ЭВМ для более подробного анализа. Взаимодействие с пользователем у модуля носит элементарный характер и производится через текстовый ЖК-дисплей с функционально-ограниченной клавиатурой.

Диагностический метод подразумевает определение амплитудно-фазовых характеристик (размах, длительность, фазовый сдвиг и др.) отраженного акустического импульса продольной либо поперечной волны. За выполнение перечисленных функций отвечает блок накопления и обработки. Блок аналитических расчетов выполняет численное моделирование распространения акустических сигналов на границе звукопровода и ДДС и может использовать полученные данные для сопоставления с экспериментальными результатами, переданными блоком накопления и обработки в банк осциллограмм. В частности, такой комплексный подход может использоваться для восстановления динамики параметров исследуемой среды методом обратной задачи.

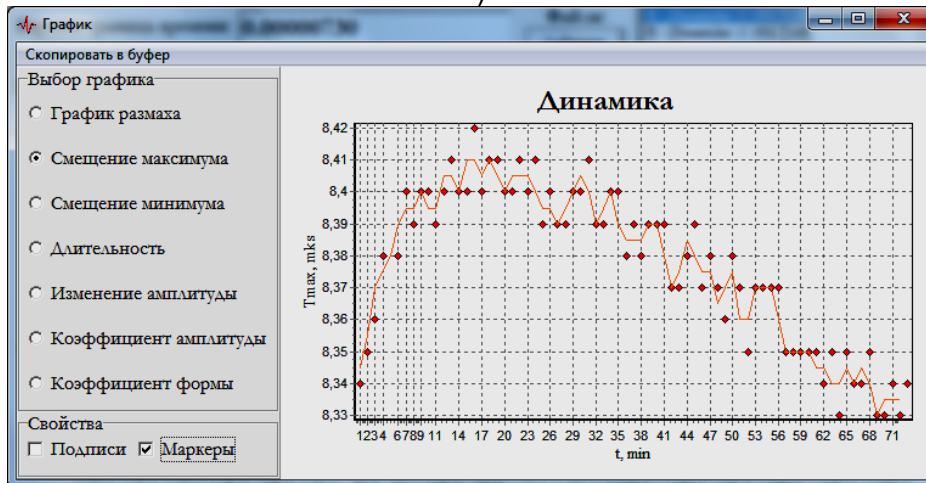


Рисунок 2 — Структура блока накопления и первичной обработки

Для решения задачи обработки осциллограмм и интерпретации динамики контролируемого процесса в ходе разработки программной подсистемы были реализованы следующие функциональные части (см. рисунок 2):



а)



б)

**Рисунок 3 — Копии экрана работающей программы:
а) – загрузка осциллограммы, б) – расчет динамики сигнала**

- Модуль основного интерфейса взаимодействует с пользователем в течение большей части времени работы программы. Пользователю предоставляется возможность загрузить файлы осциллограммы сигнала с диска (для обработки поодиночке или в составе последовательности измерений) или получить с осциллографического устройства обращением к внешнему модулю получения данных, выполнить отсечение неинформативной части сигналов путем ввода ограничения временной характеристики для процедуры фильтрации, передать последовательность осциллограмм модулю анализа динамики сигнала либо запустить модуль автоматизации измерений.

- Подсистема визуализации строит осциллограммы по данным, полученным из загруженного файла (выбранного пользователем либо созданного модулем получения данных). Кроме отрисовки отдельной осциллограммы, выбранной пользователем в списке, отображающем последовательность измерений, предусмотрена функция построения трехмерного графика для визуализации всей последовательности.

- Модуль анализа динамики сигнала реализует одну из главных задач программной системы. При переходе к этому модулю происходит вычисление необходимых интегральных характеристик сигнала и построение соответствующих графиков их изменения. В

число интегральных характеристик, поддерживаемых модулем в настоящее время, входят размах (полная амплитуда) сигнала, относительный фазовый сдвиг, коэффициент амплитуды и коэффициент формы сигнала, среднеквадратичное и среднее выпрямленное значения и некоторые другие. Предусмотрено также построение спектра сигнала.

- Модуль автоматизации измерений позволяет установить расписание опроса осциллографического устройства через модуль получения данных, а также правила отбора осциллограммы для включения в последовательность экспериментов. Расписание опроса задается в виде интервала между изменениями, в простейшем случае являющегося константой. Далее осциллограмма может быть принята в последовательность либо подвернута автоматическому анализу. Анализ предполагает временное добавление осциллограммы и построение динамики выбранных пользователем интегральных параметров. Решение об окончательном включении осциллограммы в последовательность принимается, если изменение интегрального параметра превышает указанный порог (в процентах для данной последовательности либо в абсолютных величинах). Также предусмотрен режим, при котором включение осциллограммы в последовательность сопровождается уменьшением интервала между измерениями, а отказ от включения – увеличением (вплоть до исходно заданного значения). В частности, это позволяет более эффективно исследовать процессы с неравномерной скоростью протекания.

Литература

1. Костюк, Д.А. Аномальное отражение продольного ультразвука от сильно диссипативной среды / Д.А. Костюк, Ю.А. Кузавко // Инженерно-физический журнал, 2004, т. 77, №5. – С. 161 – 169.

2. Козак, А.Ф., Костюк, Д.А., Николаюк, Л.Н. Программно-аппаратный комплекс акустического спектрального анализа диссипативно-дисперсионных сред // Проблемы проектирования и производства радиоэлектронных средств: сб-к материалов V Междунар. науч.-тех. конф., Новополоцк, 29–30 мая 2008: в 3 т. / Полоц. гос. ун-т. – Новополоцк, 2008. – Т. III. Информатика. – С. 220 – 223.

3. Костюк, Д.А., Николаюк, Л.Н. Компьютеризированная установка акустического спектрального анализа диссипативных сред // Современные информационные компьютерные технологии: сб. науч. ст. – ГрГУ // отв. ред. А.М. Кадан. – Гродно, 2006. – С. 58–63.

УДК 681.01(077.6)

ХАМРР – ПЕРСПЕКТИВА ПРИМЕНЕНИЯ

Давыденко Е.А.

УО «Белорусский государственный экономический университет», г. Минск

Много людей из своего собственного опыта знают, как непросто устанавливать веб-сервер Apache, и это становится ещё труднее, если нужно добавлять поддержку MySQL, PHP and Perl.

Хамрр — кроссплатформенная сборка веб-сервера, содержащая Apache, MySQL, интерпретатор скриптов PHP и язык программирования Perl, и много дополнительных библиотек, которые позволяют запустить полноценный веб-сервер.

Хампр — это акроним:

- **X** (любая из четырех операционных систем)
- **A**pache
- **M**ySQL
- **P**HP
- **P**erl

Полный пакет содержит:

- ✓ Web-сервер [Apache](#) с поддержкой [SSL](#)
- ✓ СУБД [MySQL](#)
- ✓ [PHP](#)
- ✓ [Perl](#)
- ✓ FTP-сервер
- ✓ [FileZilla](#)
- ✓ POP3/SMTP сервер
- ✓ утилиту [phpMyAdmin](#)

Установка Хампр

Хампр действительно очень легко установить и использовать – нужно его только загрузить и извлечь.

Хампр распространяется для четырёх платформ (операционных систем):

- Хампр для Linux
- Хампр для Windows
- Хампр для Mac OS X
- Хампр для Solaris

Для установки Хампр необходимо скачать всего один файл формата zip, [tar](#) или [exe](#), а компоненты программы не требуют настройки. Программа регулярно обновляется для включения в состав новейших версий [Apache](#), [MySQL](#), [PHP](#) и [Perl](#). Также Хампр идет с множеством других модулей, включая [OpenSSL](#) и [phpMyAdmin](#).

Пользовательский интерфейс программы настолько прост, что ее называют «сборкой для ленивых». Установка ХАМРР занимает меньше времени, чем установка каждого компонента в отдельности. Данный web-сервер распространяется в полной, стандартной и уменьшенной (известной как ХАМРР Lite) версиях. Все дополнительные модули также доступны для скачивания. Из дополнительных возможностей можно отметить, что сама компания выпускает пакеты обновления в виде zip, 7-zip, [tar](#) или [exe](#), которые позволяют обновить все компоненты с одной версии сборки Хампр на более новую.

Философия Хампр

Философия ХАМРР - построение легкой для инсталляции сборки веб-сервера для разработчиков для входа в мир Apache. Для большего удобства ХАМРР формируется со всеми включенными особенностями.

Лицензия

Хампр — компиляция бесплатного программного обеспечения (сопоставимый распределению Linux), её бесплатно и свободно можно скопировать в соответствии с [лицензией GNU General Public License](#).

Применение

Изначально Хампр создавался как инструмент для разработчиков, позволяя веб-дизайнерам и программистам тестировать свою работу не используя Интернет. Хампр сконфигурирован так, чтобы быть открытым настолько, насколько это возможно, и позволять разработчику все, что он хочет. Однако Хампр иногда используется и во всемирной паутине. Также программа поддерживает создание и управление базами данных [MySQL](#) и [SQLite](#).

Преимущества Хатрр

☑ Простая и очень быстрая установка и развертывание полноценного веб-сервера:

Для того чтобы превратить свой домашний компьютер в полноценный веб сервер и размещать в локальной сети свои сайты, совсем необязательно устанавливать и настраивать Apache, MySQL и кучу других пакетов. Есть более простой и быстрый вариант — установить Хатрр.

☑ Предусмотрена облегченная версия — Хатрр-Lite, которая требует меньше места.

☑ Хатрр — это очень удобный в управлении серверный пакет, который отвечает всем необходимым требованиям. В отличие от некоторых конкурентов он не создает никаких дисков на компьютере, иерархия папок имитирует хост.

☑ Хатрр решает все проблемы одной кнопкой; в этой сборке собраны все самые нужные и полезные функции для веб-мастеров.

☑ Главный плюс Хатрр - легкая установка.

Недостатки Хатрр

Основным недостатком считаются минимальные настройки безопасности по умолчанию. Но этот минус можно исправить, изменив их после установки. Однако если использовать Хатрр лишь периодически, запуская его для отладки веб-проектов или при отсутствии локальной сети, менять настройки необязательно.

Таким образом, на сегодняшний день Хатрр является одной из лучших сборок веб-сервера. С помощью неё можно быстро развернуть на компьютере полноценный и быстрый веб-сервер, который позволит легко создавать всё, начиная от маленьких динамических страниц, заканчивая корпоративными сайтами.

Литература

1. www.apachefriends.org – официальный сайт Хатрр
2. Петерсен, Р. Энциклопедия Linux
3. Мамаев, М., Петренко, С. Технологии защиты информации в Интернете. Специальный справочник

УДК 004.514.62

РЕАЛИЗАЦИЯ МОДЕЛИ ДИНАМИЧЕСКИХ МИНИ-ОКОН В АППАРАТНО-УСКОРЕННОМ ГРАФИЧЕСКОМ ИНТЕРФЕЙСЕ

Дёмин В.В.

УО «Брестский государственный технический университет», г. Брест

Задача манипулирования окнами в ситуации, когда их совокупная площадь существенно превышает разрешение устройства вывода, возникла почти одновременно с графическим интерфейсом пользователя. Один из ранних способов облегчить навигацию в таких условиях получил название иконификации - окно «сворачивалось» в пиктограмму соответствующего приложения, снабженную поясняющей подписью. Впоследствии концепция получила альтернативное развитие. С ростом вычислительных возможностей процессоров и разрешающей способности экранов появились графические оболочки, отображавшие вместо пиктограммы скрытого окна его уменьшенное изображение. Хотя

такой подход сопряжен с рядом технических трудностей, возможность видеть одновременно, хотя бы с уменьшенной детализацией, изображение всего рабочего пространства дает пользователю ощутимые преимущества [1, 2]. Особенно это актуально в среде динамически изменяющихся объектов, к которым относятся все многозадачные графические среды современных операционных систем.

До появления аппаратно-ускоренных графических интерфейсов вычислительные затраты, необходимые для реализации мини-окон, отображающих динамику в реальном масштабе времени, в большинстве случаев представлялись неоправданными. Разработчики ограничивались компромиссным вариантом, обновляя мини-окна через заданный промежуток времени либо вовсе оставляя в них изображение, наблюдавшееся в момент минимизации окна. Разумеется, подобные ограничения заметно снижали полезность мини-окон как модели интерфейса, а следовательно, и популярность соответствующих программных продуктов.

Одну из ранних относительно полноценных реализаций модели мини-окон в UNIX-подобных ОС можно наблюдать в оконном менеджере FVWM. В его варианте FVWM-Crystal пользователь может выбрать один из нескольких различных режимов работы (т.н. рецептов). Два из стандартных рецептов, Top-Down и Top-Line, реализуют концепцию мини-окон, называемых в FVWM тумбнейлами. При минимизации окна менеджер автоматически размещает его уменьшенное изображение в нижней части рабочего стола (рис. 1). Частое обновление тумбнейла приводит к непомерному потреблению ресурсов, поэтому интервал обновления обычно составляет 5-10 секунд (в различных вариантах настройки), либо обновление отключено.

Аналогичную функциональность для Windows XP реализуют две сторонние программы: ThumbWin и miniMize [3]. Оба приложения перехватывают события минимизации окон и располагают их уменьшенные изображения на рабочем столе. Отсутствие визуально выделяющегося обрамления часто приводит к трудностям при идентификации мини-окон среди пиктограмм. Сходство с пиктограммами усиливает также возможность перемещать мини-окна мышью по рабочему столу и реализованный в miniMize режим их случайного размещения. Обновление изображения окна не производится. Также поверх мини-окна выводится уменьшенная пиктограмма приложения для его более легкой идентификации.

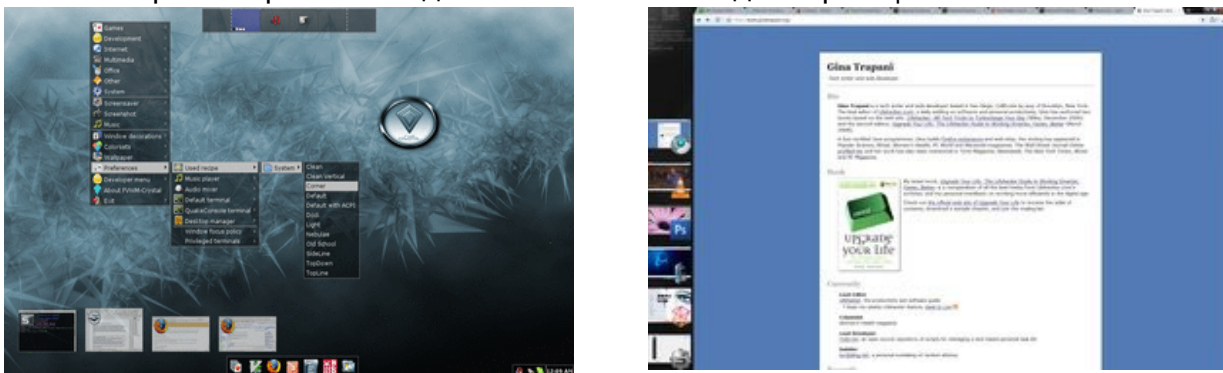


Рисунок 1 — Модель мини-окон в оконном менеджере FVWM для X Window System и в оболочке Emerge Desktop для Windows Vista

С распространением аппаратно-ускоренной графики появилась возможность реализации мини-окон с динамически обновляемым содержимым без существенного роста загрузки центрального процессора. Разработанная Sun Microsystems оболочка Looking Glass является родоначальником ряда интерфейсных решений, впоследствии заимство-

ванных в Windows Vista и MacOS X. В числе прочих панель-док в нижней части экрана демонстрирует уменьшенные изображения, соответствующие открытым окнам. Кроме того, док размещает ярлыки для быстрого запуска; отображения изображений минимизированных окон не предусмотрено. Проект Looking Glass так и не был доработан до работоспособной версии и всегда оставался демонстрационным ПО с ограниченным комплектом из четырех приложений.

Первая (и единственная на сегодняшний день) реализация динамических мини-окон под ОС Windows продемонстрирована на модифицированной конфигурации [4] графической оболочки Emerge Desktop — сторонней программы, предназначенной для замены стандартной оболочки Explorer. Боковая панель, отображающая уменьшенные изображения открытых документов, служит заменой панели задач; при наведении указателя мыши мини-окна увеличиваются в размерах, а выбор одного из них передает фокус на соответствующее окно. Отображение в реальном масштабе времени требует ОС версии Vista или 7. Панель показывает уменьшенные изображения всех окон, независимо от того, минимизированы они или нет, что в сущности нарушает рассматриваемую модель интерфейса (рис. 1).

В Windows 7 реализован технологически близкий механизм [5], отображающий уменьшенные изображения окон для выбранного на панели задач приложения. При наведении указателя мыши на кнопку соответствующего приложения в панели задач над ней отображается дополнительная панель с мини-окнами. Хотя отображение мини-окон обновляется в реальном масштабе времени, они не могут использоваться для фонового отслеживания содержимого окон, т.к. не присутствуют в видимой области постоянно и требуют от пользователя дополнительных действий для показа.

Таким образом, использование модели мини-окон долгое время сдерживалось техническими причинами, а в настоящее время ограничивается, скорее, по исторически сложившимся обстоятельствам. Однако преимущества обновляющихся в реальном масштабе времени уменьшенных изображений неактивных окон очевидны.

Нами предпринята попытка реализовать модель мини-окон в ее изначальном виде на базе аппаратно-ускоренного оконного менеджера Comriz и используемой им библиотеки OpenGL. Одно из свойств OpenGL – объекты фреймбуфера – дает оконному менеджеру эффективный доступ к окнам неактивных приложений. Для приложения объекты фреймбуфера выглядят как обычные окна, а для оконного менеджера – как текстуры, которыми можно управлять.

Разработанный модуль расширения Comriz состоит из трех функциональных частей: кода инициализации, обработчиков событий и сервисных функций.

Код инициализации вызывается при загрузке модуля. При этом системе передаются указатели на конструкторы и деструкторы, вызываемые при создании и удалении контекста дисплея, каждого из соответствующих дисплею виртуальных экранов и каждого из соответствующих виртуальному экрану окон. Доступ к контексту дисплея позволяет получать события, экран дает возможность вклиниваться в цепочку процедур отрисовки, включающую подготовительную стадию, собственно рендеринг экрана, и завершение отрисовки.

Для каждого окна предусмотрен метод PaintWindow, выполняющий обработку вывода его содержимого. Однако формирование мини-окон при минимизации окна в системе X Window System сопряжено со следующей технической проблемой. При наступлении события минимизации оконный менеджер уничтожает изображение окна и соответствующую динамическую текстуру и вновь создает ее только при восстановлении окна из

минимизированного состояния. Эта архитектурная особенность X-сервера позволяет экономить ресурсы на окнах, с которыми пользователь не работает в настоящий момент. Таким образом, поскольку показ мини-изображений таких окон невозможен, было принято решение о переназначении команды минимизации окна и закреплении ее за активированием разработанного модуля. В ходе активации запоминаются прежние координаты окна, устанавливается новое значение координат в соответствии с предназначенной для мини-окон областью внизу экрана, а также устанавливается значение коэффициента масштабирования, отличное от единицы. По этому коэффициенту код модуля, обрабатывающий события отрисовки, отличает мини-окна от стандартных окон.

При инициализации экрана модуль закрепляет собственные обработчики за необходимыми событиями. Основные из них — PreparePaintScreen, событие, предшествующее отрисовке экрана, традиционно предназначенное для предварительных вычислений; PaintWindow, в ходе которого проверяется масштаб окна и при необходимости вносятся изменения в его матрицу трансформации, и DamageWindowRect, которое определяет конкретный фрагмент экрана для перерисовки, снижая аппаратные затраты на обновление экрана.

Также служебная функция, встроенная в цикл обработки событий, отслеживает щелчок мышью по мини-окну для возвращения его к единичному масштабу и прежним координатам на экране.

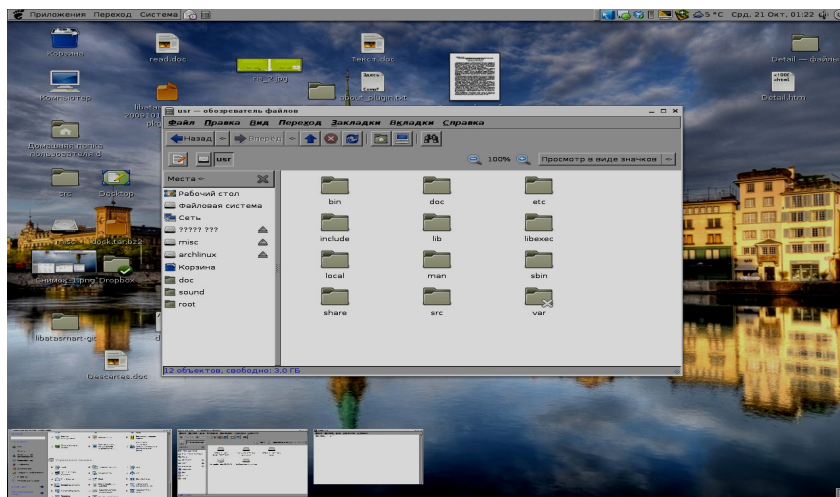


Рисунок 2 — Мини-окна, отображаемые разработанным модулем

Литература

1. Раскин, Дж. Интерфейс: новые направления в проектировании компьютерных систем / Дж. Раскин – СПб.: Символ-Плюс, 2003. – 272 с.
4. Борушко, И.Н. Применение модели периферического зрения в графическом интерфейсе пользователя. Современные информационные компьютерные технологии: сб. науч. ст. / И.Н. Борушко, Е.В. Гоманова, Д.А. Костюк – Гродно: ГрГУ, 2006. С. 22 – 27.
5. Kurdi, S. ThumbWin and miniMize: two apps that can minimize open windows as desktop thumbnails. August 25, 2008. <http://www.freewaregenius.com/2008/08/25/thumbwin-and-minimize-two-apps-that-can-minimize-open-windows-as-desktop-thumbnails/>
6. Trapani, G. Windows Vista with a Live Thumbnail Sidebar. Oct 6 2008. <http://lifelhacker.com/5058949/windows-vista-with-a-live-thumbnail-sidebar>
7. Trying out Windows 7 Release Candidate – Part 2. May 10, 2009. <http://itsbytor.wordpress.com/2009/05/10/trying-out-windows-7-release-candidate-part-2/>

УДК 004.6

ВНУТРЕННЯЯ АРХИТЕКТУРА СИСТЕМЫ ТЕСТИРОВАНИЯ E-JUDGE

Дядищев А.О., Грязных П.А.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Силаев Н.В., доцент*

Предлагаемая система тестирования решения задач программирования представляет собой сложный программный комплекс. Для облегчения его разработки и сопровождения она была разбита на несколько крупных модулей:

- Основной управляющий модуль – модуль, являющийся управляющим для остальных. Осуществляет общее управление работой системы и позволяет выполнять общие настройки;
- Сервис тестирования – позволяет работать комплексу в виде сервиса, что удобно при организации постоянно доступного тестирования;
- Модуль тестирования – модуль, осуществляющий непосредственное тестирование;
- Модуль установки комплекса – модуль, служащий для установки и первоначальной настройки комплекса;
- Визуальный редактор настроек – модуль, служащий для изменения настроек для конкретной задачи с помощью удобного графического интерфейса;
- Модуль e-Judge Contest – модуль, являющийся удобной оболочкой для работы с комплексом при организации локальных и удаленных программистских олимпиад;
- Модуль e-Judge Study – модуль, являющийся удобной оболочкой для работы с комплексом при организации лабораторных, коллоквиумов, экзаменов.

Для разработки интерфейса использовалась библиотека Microsoft Foundational Classes (MFC).

Архитектура основного управляющего модуля

Архитектурно основной управляющий модуль можно разделить на следующие составные части: отображение журнала; управление настройками; класс CTester для осуществления операций тестирования, выполненный в виде паттерна Singleton; класс CTesterSettings, являющийся контейнером для хранения настроек; класс CTesterThreads, служащий для фоновой работы процесса тестирования, а также функционирования основного управляющего модуля в режиме сервера; класс CJournal для работы с данными журнала; диалог тестирования отдельной задачи; диалог тестирования каталога с задачами.

Отображение журнала

Отображение журнала осуществляется с помощью наследника класса CListCtrlStyled. С помощью обработки события LVN_COLUMNCLICK была добавлена возможность сортировки по различным столбцам. А перехватив событие ON_WM_RBUTTONDOWN, было добавлено контекстное меню для элементов журнала.

Управление настройками

Класс COptions отвечает за реализацию диалога настроек. Для большего удобства настройки сгруппированы в виде дерева. Для реализации конкретных страниц настроек используются следующие классы: CSettingsAssociations – настройка ассоциаций для расширений сообщений; CSettingsColors – настройка цветового оформления различных записей в журнале; CSettingsExts – настройка расширений для файлов-сообщений;

CSettingsFile – пути к основным файлам; CSettingsFormats – используемые форматы сообщений; CSettingsIntFiles – настройка имен основных внутренних файлов-сообщений; CSettingsMode – настройка режима работы; CSettingsPath – настройка путей к основным требуемым папкам; CSettingsSubst – настройка подстановок для путей; CSettingsTimings – настройка производительности.

Модуль установки комплекса

Модуль установки комплекса позволяет произвести инсталляцию комплекса на компьютер пользователя и произвести первоначальную настройку. При установке имеется возможность выбора необходимых компонентов. Перечень компонентов таков: **Core files** – основные файлы системы, установка обязательна; **e-Judge Service** – компонент, необходимый при развертывании сервера тестирования, при тестировании на локальной машине не обязателен; **Examples** – примеры файлов настроек и тестов для задач; **SDK** – примеры проверяющих программ на C++ и Pascal; **Standard checkers** – стандартные проверяющие программы (сравнение файлов, сравнение дробных чисел с заданной точностью и т.д.); **Visual Tasks' Configs' Editor** – программа для визуального создания и редактирования файлов настроек для задач; **e-Judge Contest** – платформа для организации студенческих олимпиад с помощью World Wide Web; **e-Judge Study** – платформа для организации лабораторных работ, проведения коллоквиумов и экзаменов.

По завершении инсталляции в меню Пуск(**Start**) в папке **Программы** будет создана папка с двумя ярлыками, один из которых называется e-Judge и служит для запуска основной программы со стандартными настройками, а второй, Visual Tasks' Configs' Editor, запускает программу создания и редактирования файлов настроек.

Предлагаемая работа является продолжением ранее начатых исследований (2004-06 г.г.) Здановичем Д.В., за консультации которого авторы приносят благодарность.

УДК 621.374:681.511

КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ПРОЦЕССА ЦЕНТРИРОВАНИЯ ОБСАДНЫХ КОЛОНН ДЛЯ НЕФТЯНЫХ СКВАЖИН

Емельянов Д.С, Трохова Т. А.

УО «Гомельский государственный технический университет имени П. О. Сухого», г. Гомель

Применение компьютерного моделирования и автоматизации при проектировании нефтяных скважин является необходимым условием повышения качества выполняемых инженерно-конструкторских разработок, кроме того, позволяет значительно повысить сроки проектирования при строительстве нефтяных скважин. Эта задача не решена в полном объеме и не нашла до настоящего времени должной реализации в специализированных системах автоматизированного моделирования и проектирования.

Основным назначением программного комплекса является автоматизированный расчет центраторов для обсадных колонн. Центрирование обсадной колонны является одним из важнейших технологических приёмов, отсутствие которого не может быть восполнено ни одним из известных мероприятий, направленных на повышение качества крепления скважин. Центрирование способствует равномерному распределению цементного раствора вокруг

обсадной колонны, повышает качество разобращения пластов, улучшает проходимость обсадной колонны по стволу скважины и обеспечит полное замещение цементным раствором в затрубном пространстве бурового раствора при выполнении цементирования. При анализе на интервале рассматриваются изгиб обсадной колонны, степень центрирования, а также влияние плотности бурового раствора на степень центрирования.

Разработанная авторами программа дает возможность инженеру-технологу точно определить количество центраторов и расстояние между ними для обеспечения необходимой степени центрирования обсадной колонны в стволе скважины. Программный комплекс использует в качестве исходной информации общесистемные справочники и результирующие данные автоматизированного рабочего места (АРМ) «Проектные данные геолого-технического наряда» и АРМ «Расчет обсадных колонн». Программный комплекс «Расчет цементирования обсадных колонн» имеет удобный интерфейс, предназначенный для работы в нем пользователей, не имеющих профессиональной компьютерной подготовки.

Программа реализует следующие основные функции:

- ввод исходных данных для расчета процесса центрирования;
- расчёт количества центраторов на сжатом и растянутом участках обсадной колонны;
- графическая интерпретация результатов расчёта;
- ведение диалога с пользователем.

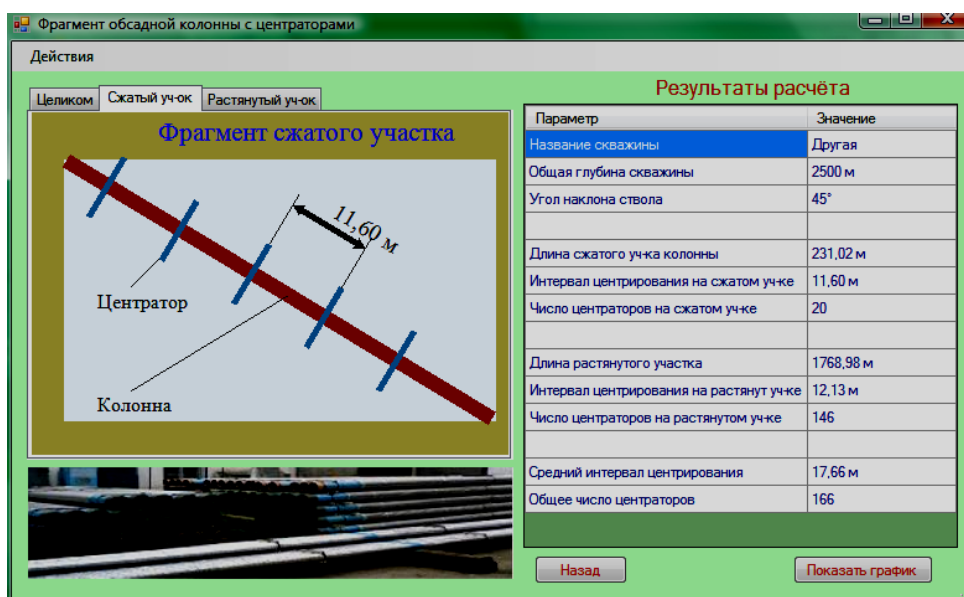


Рисунок 1 – Вид окна программы при работе в режиме сжатого участка

Программа работает в двух режимах: режим расчёта на растянутом участке обсадной колонны и режим расчёта на сжатом участке. На рисунке 1 приведён вид окна программы при работе в режиме сжатого участка. Начало работы в любом из перечисленных режимов – это выбор или ввод характеристик скважины, подлежащей расчету. Идентификация скважины может производиться несколькими способами, одним из которых является выбор скважины из «дерева», приведенного в левой части рабочего окна программного комплекса.

Программный комплекс «Расчет цементирования обсадных колонн» является важной и неотъемлемой частью автоматизированной системы инженерных расчетов в производстве работ по строительству скважин. Предполагается внедрение программного продукта в управлении буровых работ производственного объединения «Белоруснефть».

КОМПЬЮТЕРНОЕ ТЕСТИРОВАНИЕ

Жешко Н.С., Тарариев А.И.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Силаев Н.В., доцент*

Со времени появления компьютерной техники в учебных заведениях в образовании стало широко применяться компьютерное тестирование, при котором предъявление тестов, оценивание знаний учащихся и выдача им результатов осуществляется с помощью ПК.

Компьютерное тестирование может проводиться в различных формах, различающихся по технологии объединения заданий в тест.

Первая форма – компьютерная форма представления вариантов бланкового теста с фиксированным набором заданий: готовый тест, стандартизованный или предназначенный для текущего контроля, вводится в специальную оболочку, функции которой могут различаться по степени полноты.

Вторая форма – автоматизированная компоновка вариантов теста фиксированной длины из банка калиброванных заданий – предполагает автоматизированную генерацию вариантов теста, осуществляемую с помощью инструментальных средств. Варианты создаются перед экзаменом или непосредственно во время его проведения из банка тестовых заданий.

Третья форма – компьютерное адаптивное тестирование – базируется на специальных адаптивных тестах. В основе адаптивного подхода лежит индивидуализация процедуры отбора заданий теста, которая за счет оптимизации трудности заданий применительно к уровню подготовленности обучаемых обеспечивает генерацию эффективных тестов.

Среди преимуществ компьютерного тестирования можно выделить следующие:

- Компьютерное тестирование позволяет сэкономить средства, расходуемые обычно на печать и транспортировку бланковых тестов.
- При компьютерном тестировании повышается информационная безопасность и предотвращается рассекречивание теста за счет высокой скорости передачи информации и специальной защиты электронных файлов.
- Упрощается и ускоряется процедура подсчета результирующих баллов независимо от видов тестовых заданий.
- Предельно повышается объективность проверки знаний.
- При компьютерном тестировании учащемуся незамедлительно выдается тестовый балл и принимаются неотложные меры по коррекции усвоения школьником нового материала на основе анализа протоколов по результатам выполнения корректирующих и диагностических тестов.
- Повышаются информационные возможности процесса контроля, появляется возможность сбора дополнительных данных о динамике прохождения теста отдельными учащимися и для осуществления дифференциации пропущенных и недостигнутых заданий теста.
- У организатора тестирования появляется реальная возможность оперативно реагировать на степень усвоения слушателями (студентами или учащимися) изложенного в ходе занятий материала.
- Наконец, появляется возможность систематической проверки знаний обучаемых, что важно не только для проверяющего, но, в первую очередь, для проверяемых, которые в ходе подобных проверок (тестированием), как показывает практика, менее закрепощены.

При проведении компьютерного тестирования необходимо учитывать психологические и эмоциональные реакции учащихся. Негативные реакции обычно вызывают различные ограничения, которые традиционно накладываются при выдаче заданий при тестировании «на твердой основе». В ходе компьютерного тестирования подобные негативные моменты можно, если не убрать совсем, то хотя бы смягчить. Например, фиксируется либо порядок предъявления заданий, либо максимально возможное время выполнения каждого задания, после истечения которого, независимо от желания испытуемого, появляется следующее задание теста. Учащиеся при адаптивном тестировании бывают недовольны тем, что не имеют возможности пропустить очередное задание, просмотреть весь тест до начала работы над ним и изменить ответы на предыдущие задания. Иногда школьники возражают против компьютерного тестирования из-за трудностей, которые возникают при выполнении и записи математических вычислений и т.д.

Для снижения влияния опыта работы учащихся с компьютером на тестовые баллы рекомендуется включать в оболочки для компьютерного тестирования специальные инструкции и тренировочные упражнения для каждой инновационной формы заданий. Необходимо также предварительно ознакомить учащихся с интерфейсом программы, провести репетиционное тестирование и выделить в самостоятельные группы учащихся, не имеющих достаточного опыта работы с ПК, для того чтобы дополнительно обучить их или дать им бланковый тест.

Таким образом, компьютерное тестирование выступает как инструмент управления учебным процессом, как элемент обратной связи, который дает возможность анализировать учебный процесс, вносить в него коррективы, т.е. осуществлять полноценное управление процессом обучения. Систематическое использование компьютерных тестов в качестве промежуточного контроля успеваемости определяет учебный процесс как систему непрерывного контроля и самоконтроля учащихся, которая дает возможность учителю получать «обратную связь», а учащимся — возможность в течение всего обучения отслеживать уровень своей подготовленности, а самое основное – интенсифицировать учебный процесс, привести его в соответствие нашему быстротекущему времени.

УДК 65В631

МОДЕЛЬ РАСЧЕТА ЭКОНОМИЧЕСКОГО РАЗДЕЛА ДП

Игнатенко В.Ю.

УО «Барановичский государственный университет», г. Барановичи

ВВЕДЕНИЕ. В современных условиях сформировался принципиально иной подход к экономической подготовке научно-технических кадров. Подготовка специалистов в университетах страны в новой экономической среде обусловлена повышенными требованиями как к их знаниям в рамках определенной номенклатуры специальностей, так и к расширению их общего кругозора, в котором экономической подготовке отводится важное место. Это проявляется в прохождении и усвоении определенных учебных курсов по экономике, организации производства и управлению и завершается изложением материалов в экономической части дипломного проекта, в которой полученные знания преломляются через призму конкретной тематики, представленной в соответствии с профессиональной направленностью системотехническими, программно-математическими, конструкторско-техническими или технологическими решениями.

В этой связи при разработке и защите дипломного проекта, на завершающей стадии подготовки высококлассного специалиста по программному обеспечению, формулируются требования к прикладному применению полученных экономических знаний путем исполнения экономической части, в которой должны быть продемонстрированы аналитические способности к истолкованию и использованию экономических факторов в конкретной сфере приложения [1].

Автором была разработана программа для расчета экономического раздела дипломного проекта по базовому и проектному варианту.

ОСНОВНАЯ ЧАСТЬ. Автором создана программа для расчета экономического раздела дипломного проекта по базовому и проектному варианту [2]. В программе предусмотрены следующие возможности:

- Ввод пользователем данных и требуемых для расчета коэффициентов (интервал ввода коэффициентов контролируется программно).
- Обработка исключительных ситуаций (при вводе нечисловой информации выводится сообщение об ошибке).
- Предусмотрена организация промежуточных расчетов в виде отчета.
- Организация представления конечных результатов в виде таблицы.

Весь расчет экономического раздела разбит на подзадачи, о чем свидетельствует наличие вкладок. Переход к следующей вкладке доступен только тогда, когда рассчитана предыдущая вкладка либо нажата клавиша «Далее». При запуске программы пользователю предлагается ввести входные данные и необходимые коэффициенты (рисунок 1) на вкладке, затем нажимается клавиша «ОК», и так до тех пор, пока не рассчитается весь раздел.

Входные данные:	Подбор коэффициента	Результаты:
Число операторов в программе: 6	Тоа= 75	Трудоемкость отладки: 2,2
Коэффициент увеличения: 1,2	Тбс= 20,9	Трудоемкость программы: 4,22394246790982
Коэффициент коррекции: 0,4	Тп= 21,2	
	Тотл= 4,2	
	Тдр= 16,2	

Рисунок 1- Внешний вид приложения

Так как в расчете были учтены все требования к исходным данным (например, ввод нечисловой информации), то при неправильном вводе будет выведено сообщение об ошибке или о неправильном вводе коэффициента, как представлено на рисунке 2.

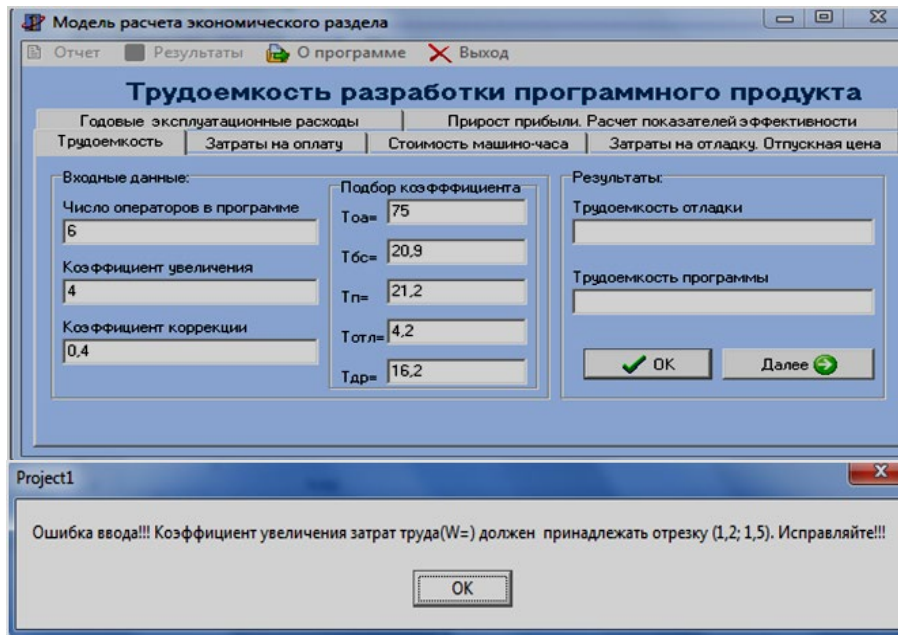


Рисунок 2 – Коррекция ошибки ввода данных

Конечный результат работы программы можно увидеть только тогда, когда будет проведен весь расчет. Он представлен в виде таблицы на рисунке 3, а все промежуточные расчеты содержатся в отчете, после проведения всего экономического расчета, который представлен на рисунке 4.

Наименование показателя	Базовый(ручной)	Проектный
Трудоёмкость решения задачи		4,22394246790982
Периодичность решения задачи		4
Годовые текущие затраты, руб		292,63865450875
Отпускная цена программы, руб		1,33333333333333
Число возможных покупателей		5
Прирост усл. прибыли, руб./год		222,40537742665
Годовой эконом. эффект, руб		221,60537742665
Срок возврата инвестиций, лет		0,0239802355277686

Рисунок 3 – Таблица результата программы

The screenshot shows a report window titled "Отчет" (Report) with the following content:

1 Определение трудоёмкости разработки программного продукта
 Условное число операторов Q=9,24
 Трудоёмкость описания задачи Тоа=0,14784
 Трудоёмкость разработки блок-схемы Тбс=0,442105263157895
 Трудоёмкость программирования по готовой блок-схеме Тп=0,435849056603774
 Трудоёмкость отладки программы на ЭВМ Тотл=2,2
 Трудоёмкость подготовки документации по задаче Тдр=0,57037037037037
 Трудоёмкость редактирования, оформления по задаче Тдо=0,427777777777778
 Трудоёмкость разработки программного продукта Трз=4,22394246790982

2 Определение себестоимости создания ПП
 2.1 Определение затрат на оплату труда разработчика
 Среднечасовая ставка работника tчр=1352
 Затраты на оплату труда разработчика Зрз=34827

2.2 Определение стоимости машино-часа работы ЭВМ
 Расходы на энергию за час работы ЭВМ, руб Се=340,2
 Годовая величина амортизационных отчислений на реновацию ЭВМ Азвм=192,276
 Годовые затраты на ремонт и техническое обслуживание ЭВМ Рзвм=0,42
 Годовая величина амортизационных отчислений на реновацию площадей Апл=5,46
 Годовые затраты на ремонт и содержание производственных площадей Рпл=3,9
 Балансовая стоимость ЭВМ, руб Цзвм=4,2

Рисунок 4 – Отчет

ЗАКЛЮЧЕНИЕ. В ходе исследования была разработана программа для расчета экономического раздела дипломного проекта по базовому и проектному варианту. Среда разработки C++Builder.

Системные требования:

- Дисковое пространство - 4 Мб;
- Занимаемая оперативная память – 5Кб;
- Операционная система –Windows 2000, XP или VISTA.

Данная работа рекомендуется в качестве методического обеспечения для студентов 5 курса специальности «Информационные системы и технологии».

Основные достоинства:

- Построена модель экономического раздела;
- Создано приложение для ее реализации;
- Произведены расчеты по базовому и проектному варианту.

К недостаткам можно отнести:

- Сложная математическая модель.

Литература

1. Кубенский, А.А. Структуры и алгоритмы обработки данных: объектно-ориентированный подход и реализация на С++ / А.А. Кубенский – СПб.: БХВ-Петербург, 2004. – 464 с.: ил.

2. Методические рекомендации по написанию экономического раздела дипломного проекта для специальности 1-40 01 02 «Информационные системы и технологии» / Под ред. Д.А. Лабоцкий, Ю.Е. Горбач.

УДК 004.658

ИНФОРМАЦИОННО-ПОИСКОВАЯ СИСТЕМА “НАУЧНЫЙ ПОТЕНЦИАЛ ВУЗА”

Копылов Д.А., Никонюк А.Н.

*УО «Брестский государственный технический университет», г. Брест
Научный руководитель В.М.Ракецкий, к.ф.-м.н, доцент*

Введение. Базы и банки данных широко проникли в нашу повседневную жизнь. Их использование позволяет собирать, накапливать, систематизировать и обрабатывать информацию в самых различных сферах человеческой деятельности, что в общем случае ускоряет подготовку и повышает качество принимаемых решений. Поисковые системы, построенные на основе баз данных (БД), за мгновение находят и представляют в систематизированном виде информацию, на поиск которой в обычных условиях уходят часы, а то и дни рабочего времени.

Одним из важнейших направлений в деятельности высшего учебного заведения является научно-исследовательская деятельность. Научный потенциал вуза является предметом интереса многих управленческих органов. Работникам научно-исследовательских подразделений достаточно часто приходится заниматься подготовкой различных отчетов, в которых представляются сведения не только за отчетный период, но и в сравнении с предыдущими периодами деятельности. Да и для внутреннего использования систематизированная информация о научно-исследовательской деятельности подразделений и отдельных работников является самостоятельной ценностью. Поэтому разработка базы данных в этой предметной области представляет интерес и востребована временем.

В настоящей работе описан один из подходов к разработке информационно-поисковой системы (ИПС), которая получила название «Научный потенциал вуза». Эта система разработана авторами по заданию научно-исследовательской части УО «Брестский государственный технический университет» и в настоящее время находится в стадии внедрения.

Концептуальная модель базы данных. На основе изучения и анализа предметной области, учета требований и пожеланий работников НИЧ БрГТУ был сделан вывод о том, что двумя основными сущностями базы данных «Научный потенциал вуза» являются:

- преподаватели и сотрудники вуза, занимающиеся научно-исследовательской деятельностью;
- задачи экономики страны в целом и региона в частности (так называемый задачник промышленности).

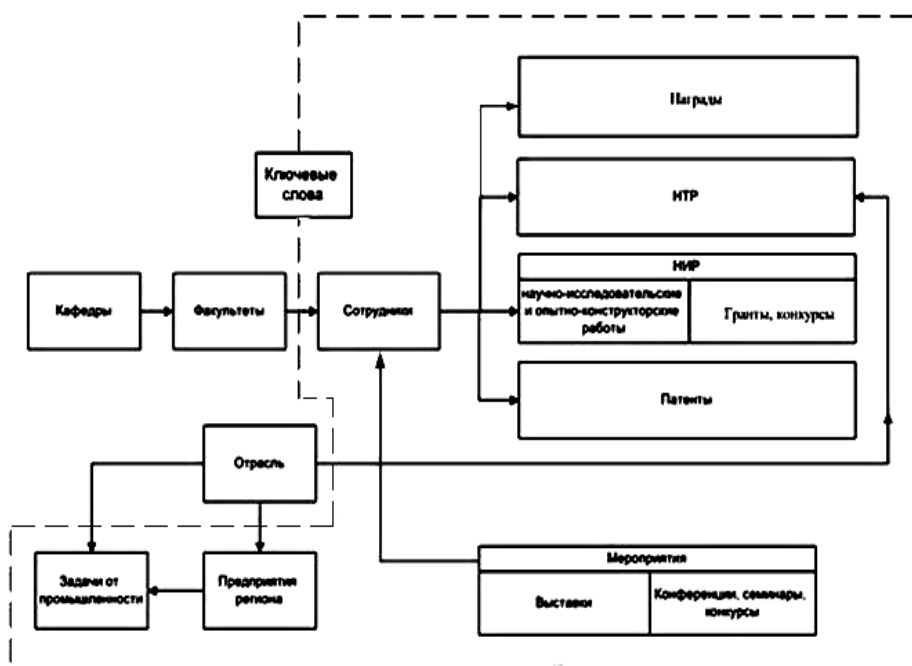
Такое сочетание позволяет не только собирать информацию о научном потенциале вуза, но вырабатывать рекомендации по его использованию в целях повышения эффективности экономики.

Научный потенциал работников вузов предполагается описывать с помощью следующих характеристик:

- научно-технические разработки, в создании которых они принимали участие;
- госбюджетные и хоздоговорные НИР, в которых принимают или принимали участие работники вуза (включая гранты и конкурсы различных организаций);
- патенты на изобретения, полезные модели, промышленные образцы;
- конференции, семинары, конкурсы, выставки и прочие научно-технические мероприятия, в проведении которых принимали участие работники вуза;
- награды, полученные работниками на различных выставках, конкурсах и т.п.

С целью структуризации и систематизации информации в БД включена информация о структуре вуза (факультеты, кафедры) и структуре народного хозяйства страны (отрасли, предприятия).

Важным элементом БД являются ключевые слова, которыми могут сопровождаться как основные сущности, так и второстепенные. Именно с помощью ключевых слов может осуществляться отбор информации по различным запросам как внутри сущностей, так и в их объединении.



Общая схема разработанной базы данных представлена на рисунке 1.

Рисунок 1 – Концептуальная модель базы данных

Особенности интерфейса ИПС. Интерфейс программы удобный и интуитивно понятный. Авторы стремились к этому, но понятие удобства – вещь субъективная, и интуиция у всех работает по-своему, поэтому, вполне возможно, что разработанный интерфейс будет кого-то чем-то не удовлетворять. В любом случае, как бы ни был построен интерфейс, всегда найдутся те, кого он будет не устраивать.

Главное окно ИПС приведено на рисунке 2. Оно предназначено для отображения информации о двух основных сущностях ИПС – сотрудниках (как показано на рисунке) и задачах промышленности. Переключение между сущностями происходит с помощью специальной кнопки на панели инструментов (на рисунке на ней надпись «Предприятия»).

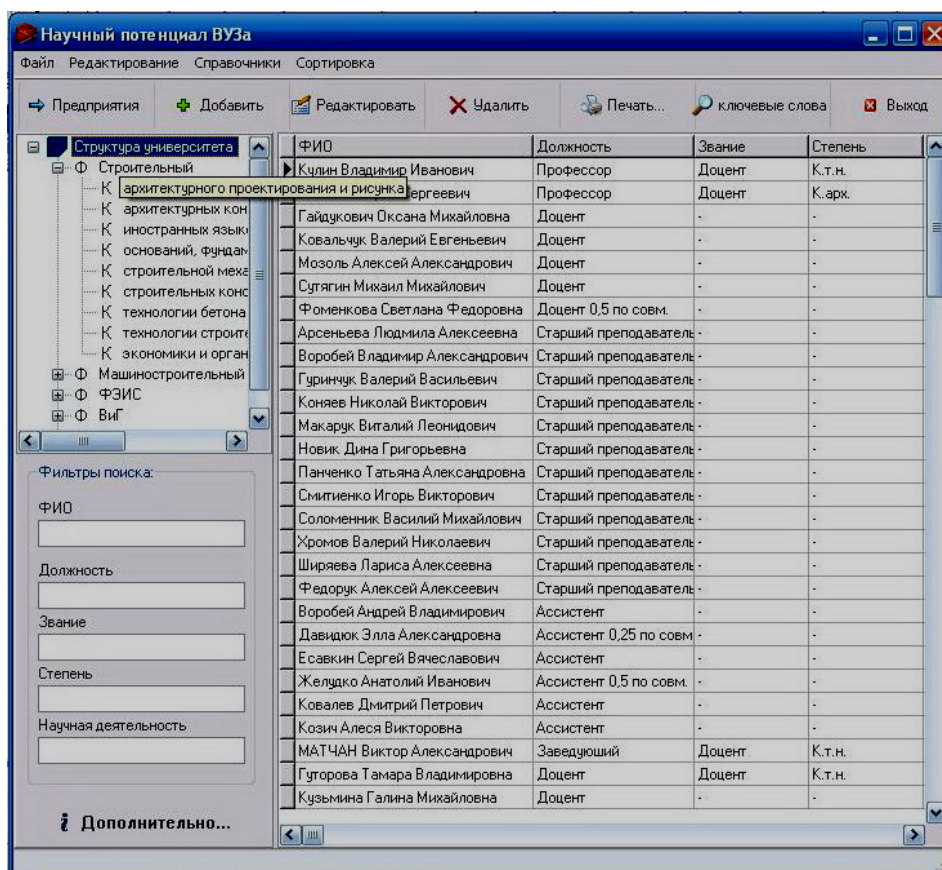


Рисунок 2 – Главное окно программы

Окно содержит два основных элемента интерфейса – дерево выбора и информационную таблицу. Дерево выбора отображается слева, имеет несколько узлов и может быть изменено по размерам мышью или полностью погашено. Для его построения используется информация о структуре вуза либо об отраслях народного хозяйства.

В верхней части главного окна расположена панель инструментов, позволяющая выполнять:

- переключение между главными сущностями;
- наиболее часто применяемые операции над записями: добавление, редактирование, удаление;
- печать выходных отчетов;
- поиск по ключевым словам.

В левом нижнем углу окна размещена панель для установки фильтра на информацию, содержащуюся в таблице. С ее помощью можно осуществлять быстрый поиск нужной информации по нескольким критериям.

Меню программы дублирует некоторые команды панели инструментов, позволяет сортировать информацию, представленную в таблицах, и получать доступ к другим таблицам БД (пункт «Справочники»), в частности, к таблицам, содержащим информацию о структуре вуза и промышленности.

Доступ к информации, характеризующей научный потенциал работников вуза (научно-технические разработки, патенты и др.), реализован двумя способами. Во-первых, через команды в меню «Справочники, во-вторых, с помощью кнопки «Дополнительно» (на рис. 2 она расположена снизу слева). В первом случае пользователь имеет доступ ко всем данным одной таблицы, которую он выбрал в меню (например, данным таблицы «Научно-технические разработки»). Во втором случае на экране появляется окно, на вкладках которого отображаются все сведения обо всех составляющих научного потенциала одного конкретного работника, на котором находился курсор в главной форме.

Остановимся подробнее на поиске информации по ключевым словам. Объектом поиска могут быть: 1) главные сущности системы; 2) подчиненные сущности; 3) главные сущности, которые содержат искомые ключевые слова или которым подчинены сущности, содержащие искомые слова. Интерфейс программы позволяет произвести необходимую настройку поиска. При этом в последнем случае (случае 3) поиск реализуется в два прохода: сначала находятся подчиненные сущности с необходимыми ключевыми словами, потом анализируются главные сущности (с учетом найденных подчиненных сущностей).

Реализация. Разработанная база данных реализована в рамках СУБД Access [1,2]. Программное обеспечение информационной системы разработано на языке Object Pascal в системе программирования Delphi 7[3,4].

Литература.

1. Кузин, А.В. Разработка баз данных в системе Microsoft Access: учебник / А.В. Кузин, В.М. Демин – М.: ФОРУМ, ИНФРА-М, 2005. – 224 с.
2. Кошелев, В.Е. Access 2007. Эффективное использование / В.Е. Кошелев – М: БИНОМ. Лаборатория знаний, 2008. – 592 с.
3. Архангельский, А.Я. Программирование в Delphi: учебник по классическим версиям Delphi / А.Я Архангельский – М: БИНОМ, 2006. – 1152 с.
4. Фаронов, В.В. Delphi. Программирование на языке высокого уровня: учебник для вузов / В.В. Фаронов – СПб: Питер, 2009. – 640 с.

УДК 37:004

MOODLE КАК ОСНОВА ОРГАНИЗАЦИИ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ МАТЕМАТИКЕ

Корольчук Д.В.

УО «Брестский государственный университет имени А.С. Пушкина», г. Брест

Необходимость использования современных информационных технологий в процессе обучения отражена в постановлении Министерства образования Республики Беларусь (№ 129 от 26.12.2006) по разработке электронных образовательных ресурсов для системы образования Республики Беларусь. Как следствие, повышение спроса на электронное учебное оборудование и программное обеспечение. Актуализируется вопрос оптимального выбора информационной технологии для улучшения качества образования. Необходимо определить цель использования технологий в учебном процессе, что определяется их дидактическими функциями.

В рамках исследования «Методика организации управляемой самостоятельной работы студентов-заочников средствами центра дистанционного обучения математике» мы используем авторскую модель центра дистанционного обучения математике (<http://www.yznaica.ru>), основанную на интеграции очных и дистанционных форм обучения. Представим данную модель в виде схемы (Рисунок 1).



Рисунок 1 – Модель центра дистанционного обучения математике

С помощью компьютерных телекоммуникаций мы оптимизируем процесс обучения студентов заочной формы обучения. Это происходит благодаря тому, что выбранные средства (компьютер, сеть, гипертекст, форум и др.) обладают следующими дидактическими свойствами [1]:

- 1) оперативная передача любого вида информации на любое расстояние;
- 2) возможность длительного хранения информации, а также ее редактирования, распечатки, обработки и т.п.;
- 3) возможность интерактивности;
- 4) возможность доступа к различным источникам информации и др.

Эти достоинства гармонично воплощены в дистанционном обучении на базе Интернет-ресурса. Соглашаясь с Е.С. Полат, отметим, что дистанционное обучение «базируется на трех составляющих: технологической, содержательной и организационной» [2, с. 44]. Каждая из перечисленных составляющих требует материальных и интеллектуальных ресурсов. Рассмотрим более подробно технологическую составляющую дистанционного обучения. В него входят затраты на оборудование, программное обеспечение, техническая поддержка, администрирование и др. Особо сложным является выбор системы дистанционного обучения (СДО). Ранее были сформулированы требования к таким системам [3, с. 282]. Большинство СДО являются платными, например ПРОМЕТЕЙ, но достоинством таких разработок является безоговорочная техническая поддержка. В рамках проводимого исследования мы остановили выбор на СДО MOODLE (официальный сайт разработчика: <http://www.moodle.org>). Moodle (modular object-oriented dynamic learning environment – модульная объектно-ориентированная динамическая учебная среда) – бесплатная, открытая (Open Source, под лицензией GNU Public License) система управления обучением (LMS), которая переведена на десятки языков и используется в более чем двухсот странах мира. Основателем и разработчиком данной системы является Martin Dougiamas из Австралии, но в силу «открытости разработки», в данный момент над системой трудятся программисты всего мира. Среда написана на языке PHP с использованием SQL-базы данных (MySQL, PostgreSQL, Microsoft SQL Server и др.).

Достоинством данной системы является, во-первых, модульная архитектура, которая позволяет внедрять, без ущерба основной системе новые расширения для разработки учебных материалов (например, Book, Wikipedia Block, Flash Video и др.). Во-вторых,

существует локальная версия Moodle, что позволяет вести разработку учебных материалов без доступа в Интернет. В-третьих, существует возможность настроить систему «под себя», в том числе и внешний вид.

Для оптимизации процесса обучения студентов-заочников мы можем использовать такие возможности, как: глоссарий, ресурс, задание, форум, wiki, урок, тест и др. Для более удобной навигации по курсам в СДО Moodle создана двухуровневая архитектура категория–курс. Так, например, нами создана категория «Курсы для студентов заочной формы обучения», в которой размещаются курсы по темам. Список всех доступных курсов можно просмотреть на странице Категории курсов (Рисунок 2).

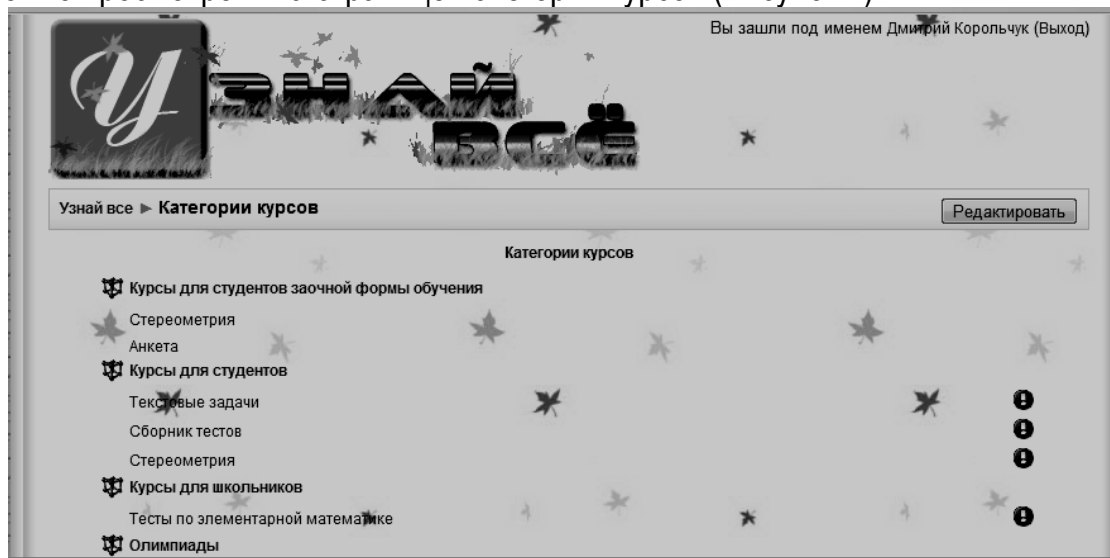


Рисунок 2 – Категории курсов

Контроль уровня знаний нами осуществляется с помощью ресурсов «тест» и «задание». Можно создавать тесты с вопросами различных типов: вопросы в закрытой форме (множественный выбор); Да/Нет; короткий ответ; числовой; соответствие; вложенный ответ и др. С помощью заданий вы формулируете задачу и задаете вариант сдачи решения: ответ – в виде нескольких файлов; ответ – в виде текста; ответ – в виде файла; ответ – вне сайта. Все полученные отметки хранятся в индивидуальном портфолио, там же вы можете посмотреть время изучения курса, выполнения задания и др., то есть контролировать процесс самообучения студента. Ресурс «форум» и «чат» предоставляют возможности для проведения консультаций, конференций и т.п. Это позволяет создать обсуждение вокруг некоторой проблемы, которая формулируется преподавателем либо студентом, благодаря этому процесс обучения становится динамичным и интерактивным.

Нами перечислены самые важные возможности СДО Moodle, в которых выражены дидактические свойства компьютерных технологий.

Литература

1. Дистанционное обучение: учеб. пособие для пед. вузов / Е.С. Полат, М.В. Моисеева, А.Е. Петров [и др.]; под ред. Е. С. Полат. – М.: ВЛАДОС, 1998. – 192 с.
2. Теория и практика дистанционного обучения: учебное пособие для студентов вузов, обучающихся по педагогическим специальностям (ОПД. Ф. 2-Педагогика) / Е.С. Полат, С. А. Бешенков, М. Ю. Бухаркина ; ред. Е. С. Полат. – М.: Академия, 2004. – 416 с.
3. Корольчук, Д. В. Из опыта организации центра дистанционного обучения математике / Д. В. Корольчук // НИЭУП: Молодежь и наука: реальность и будущее. Материалы II Международной науч.-практ. конф. – Невинномысск, 2009. – С. 281–283.

СРАВНИТЕЛЬНЫЙ ОБЗОР «БЫСТРОГО» АЛГОРИТМА ГЕНЕРАЦИИ ГРАФОВ**Кочурко В.А.**

УО «Брестский государственный технический университет», г. Брест

Введение. Задача создания полного списка регулярных графов, вместе с задачей распознавания изоморфных графов, является одной из старейших задач конструктивной комбинаторики. Помимо простого научного интереса, генерация таких графов имеет также и практическое значение.

Регулярные графы применяются в разнообразных областях, например, в peer-to-peer сетях, некоторые из которых базируются на связных регулярных неориентированных графах, при идентификации молекул веществ, при разработке баз знаний интеллектуальных систем или при создании чипов и микросхем.

В данной работе анализируется так называемый «быстрый» алгоритм, предложенный профессором Баварского университета Маркусом Мерингером в [1], и сравнивается с алгоритмом двух вершин, предложенным в [2]. Анализ базируется на программных реализациях этого алгоритма, созданном коллегами господина Мерингера, и алгоритма двух вершин, созданном в Брестском государственном техническом университете.

Постановка задачи. Регулярный граф $\tilde{A}_N^{\hat{E}}$ - граф, у которого степени всех вершин равны между собой и где N – количество этих вершин, а K – степени вершин. Задача состоит в генерации всех неизоморфных между собой регулярных графов с заданными K и N .

Переформулируем задачу в терминах теории групп. Пусть G_N - множество простых помеченных графов с множеством вершин $\{1, \dots, n\}$. Подмножество из K -регулярных графов обозначим $R_{N,K}$. Опишем граф через множество его ребер:

$$\tilde{A} = \{e_1, e_2, \dots, e_t\} \subseteq \binom{\{1, \dots, n\}}{2} =: X_n$$

Если $e = (v, w) \in X_n$ обозначает ребро, то всегда принимается, что $v < w$. Симметрическая группа S_N действует на основе множества X_n и, тем самым вызывает действия на G_N и $R_{N,K}$.

$S // G_N$ и $S // R_{N,K}$ обозначают орбиты этих действий. Задача состоит в вычислении набора представителей орбиты $S // R_{N,K}$.

Краткий обзор «быстрого» алгоритма. Для начала упорядочиваем полное множество ребер X_n следующим образом: Для $e = (v, w), e' = (v', w') \in X_n$ определим отношение «больше» - $e < e' := v < v' \vee (v = v' \wedge w < w')$.

Тогда, пользуясь этим отношением, упорядочим все ребра в графах Γ . Аналогично, упорядочим все графы Γ в множестве графов G_n , пользуясь отношением «больше» для графов:

$$\tilde{A} = \{e_1, e_2, \dots, e_t\}, \tilde{A}' = \{e'_1, e'_2, \dots, e'_t\}$$

$$\tilde{A} < \tilde{A}' :=$$

$$(\exists i \leq \min\{t, t'\} : e_j = e'_j \forall j < i \wedge e_i = e'_i)$$

$$\vee (t < t' \wedge e_j = e'_j \forall j \leq t)$$

В статье [1] показано доказательство необходимости поиска канонических представителей $-rep_{<}(S // R_{N,K})$ - орбиты $S // R_{N,K}$ как минимальные в их орбите. Так же доказана теорема, по которой, если граф Γ является представителем минимальной орбиты, то и граф Γ_1 , где $\tilde{A}_1 < \tilde{A}$ и $\tilde{A}_1 \subset \tilde{A}$, также является представителем минимальной орбиты. Базируясь на этой теореме, формулируем рекурсивный алгоритм перебора на псевдоязыке - процедура $P(\Gamma)$:

1. Точка входа: $P(\{1,2\})$.
2. Проверка: может ли входной аргумент – граф Γ – быть распространён на K -регулярный граф с N вершинами, если нет – выход.
3. Проверка: выполняется ли условие $\tilde{A} \in rep_{<}(S // R_{N,K})$, если нет – выход.
4. Если $\tilde{A} \in R_{N,K}$: добавить Γ в выходное множество, выход.
5. Для каждого $e \in X_n$ с $e > \max\{e' \in \tilde{A}\}$ вызывают $P(\Gamma + \{e\})$ с увеличением e .

Очевидно, что трудоёмкость и используемая память алгоритма в таком виде несравнимо более велики, чем для алгоритмов, описанных в [2].

Уменьшение этих параметров происходит за счёт использования полуэмпирических лемм Грюнда и Бринкманна, которые подробно рассмотрены в [1].

Используемая память:

$$O\left(\left(\frac{N * (N - 1)}{2}\right)^{\frac{NK}{2}}\right) \Rightarrow O(N^{NK})$$

Краткий обзор алгоритма двух вершин. Алгоритм двух вершин был подробно рассмотрен и сравнен с другими алгоритмами в [2] и является, возможно, лучшим представителем семейства алгоритмов, базирующихся на обработке исключительно матриц смежности. Его трудоёмкость меньше, а список генерируемых графов полон.

Допустим, что существует исходный граф \tilde{A}_n^k , из которого необходимо получить граф \tilde{A}_{n+2}^k . Тогда возьмем ребро (x_i, x_j) этого графа, и установим на нем две вершины x_{n+1} и x_{n+2} . Эти вершины будут смежны между собой и смежны вершинам x_i и x_j .

Вторым шагом является удаление ребер. Поочередно перебираются все ребра, неинцидентные вершинам x_i и x_j , и в каждом случае такое ребро (x_{i1}, x_{j1}) удаляется. В таком случае появляется возможность установки новых ребер, причем двумя способами: (x_i, x_{j1}) и (x_{i1}, x_j) , или же (x_{i1}, x_i) и (x_j, x_{j1}) . Полный перебор всех ребер в качестве входных завершает данный алгоритм.

Трудоёмкость данного алгоритма:

$$\frac{(M - 2)N(N - 1)}{2} \Rightarrow O(MN^2)$$

Количество используемой памяти:

$$O((K * N)^{K-1}) \Rightarrow O((KN)^K)$$

Программное сравнение. Главной сложностью в попытке сравнения алгоритмов является сложность определения трудоёмкости быстрого алгоритма из-за использования большого количества исключений и нечёткого обозначения алгоритма проверки принадлежности графа множеству канонических представителей; следовательно, сравнению по главному количественному критерию – трудоёмкости – алгоритмы не подлежат. Сравне-

ние же верхних оценок используемой памяти даёт поверхностное представление о чуть большей компактности алгоритма двух вершин. Однако попробуем сравнить результаты программных реализаций этих алгоритмов, для чего приведём таблицу таковых:

N	K	Кандидаты (Быстрый)	Кандидаты (2x вершин)	Время, с (Быстрый)	Время, с (2x вершин)
10	3	37	84	0	0
12	3	214	672	0	0,3
14	3	1406	7208	0,1	2
16	3	10432	49874	1	18,3
18	3	96279	-	10,8	-
20	3	1079585	-	139	-

Испытание проводилось на одинаковой компьютерной конфигурации. Графа «Кандидаты» для каждого алгоритма – это количество рабочих графов перед последней стадией отсеивания. Также стоит упомянуть, что на выходе быстрого алгоритма – список неизоморфных между собой графов, в то время как к выходному списку алгоритма двух вершин следует применить ещё и любой алгоритм изоморфного отсеивания, что увеличивает в разы трудоёмкость и, соответственно, время работы.

Заключение. Таблица 1 ясно указывает на сильный перевес в производительности для быстрого алгоритма, в то время как единственным его слабым местом является верхняя оценка количества используемой памяти. На общую оценку алгоритма в наше время больших объёмов памяти это уже не влияет так существенно, как могло бы ещё лет 20 назад – алгоритм оправдывает своё название.

Литература

1. Meringer, M. Fast Generation of Regular Graphs and Construction of Cages. Journal of Graph Theory 30, 137-146, 1999.
2. Кочурко, В. А. Некоторые алгоритмы генерации графов. Сборник конкурсных научных работ студентов и магистрантов / В.А. Кочурко – БрГТУ, 2009. – Ч. 1. – С.163-167.
3. Шуть, В. Н. Генерация регулярных связных графов / В.Н. Шуть, В.М. Свирский, Г.Л. Муравьев, С.В. Анфилец / Вестник БрГТУ. – 2006. – №5: Физика, информатика, математика. – С. 44 – 47.

УДК 372.016:51

ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ДЛЯ АДАПТАЦИИ УРОВНЯ МАТЕМАТИЧЕСКОЙ ПОДГОТОВКИ СТУДЕНТОВ-ИНОСТРАНЦЕВ

Крагель Е.А.

УО «Брестский государственный технический университет», г. Брест

Многовекторность внешней политики Беларуси делает ее привлекательной с точки зрения инвестиций и не только материальных. В настоящее время наметилась тенденция увеличения числа иностранцев, обучающихся в отечественных вузах. Последнее обусловлено следующими факторами:

- стабильная политика социальной ситуации в стране (отсутствие конфликтов на межнациональном и религиозном уровне);
- низкая стоимость обучения (оплата за год обучения в вузе для иностранцев превышает в несколько раз плату за обучение отечественных студентов, тем не менее эта сумма приемлема по сравнению с общемировыми стандартами);

- достаточно высокое качество образования;
- возможность дальнейшего трудоустройства в совместных предприятия;
- комфортные условия обучения (предоставление общежития, проведение специальных языковых курсов для адаптации).

Сказанное находит отражение в следующем факте. За последние два года количество обучаемых иностранцев выросло на 10 – 15%, что составляет 7 270 иностранцев из 83 стран мира (России, Китая, стран Ближнего Востока и Азии). Большинство иностранных студентов обучаются в медицинских институтах, технических университетах, не является исключением и Брестский государственный технический университет.

Проблеме обучения студентов-иностранцев посвящены ряд работ педагогов, методистов, среди которых Л. Т. Мазитова, Н.К. Маяцкая, Вэй Син, И.Ю. Смелкова, С.В. Копытько, Л.А. Апанасюк, Е.Т. Хачатурова, Л.С. Асейкина, Н.А. Ильенко, Г.М. Левина, Н.А. Мишонкова.

Ретроспективный обзор литературы по теме исследования показывает, что:

- большинство работ имеет узкую специализацию (например, обучению русскому языку);
- выделяются работы воспитательного аспекта (на основе сравнительного анализа вредных привычек студентов до и в период обучения в Республике Беларусь);
- многие работы являются кусочными, так как не содержат общеприменимых выводов;
- отсутствует исследование по комплексной адаптации студентов-иностранцев к обучению в технических вузах.

Сказанное определяет актуальность темы исследования: «Методика адаптации уровня математической подготовки студентов-иностранцев к требованиям технического вуза (на примере подготовительного отделения)».

Тезисно выделим основные идейные линии исследования.

Во-первых, уровень подготовки абитуриентов-иностранцев и отечественных абитуриентов кардинально отличается (как правило, у иностранцев он не достаточно высок).

Во-вторых, возникает проблема выравнивания (адаптации) уровня математической подготовки абитуриентов-иностранцев к обучению в техническом вузе.

В-третьих, необходимо решать проблему языкового барьера не только с помощью языковых курсов, но и с помощью математических курсов для усвоения математической терминологии.

В-четвертых, помимо аудиторной работы, должна быть организована внеаудиторная самостоятельная работа абитуриентов, основанная на принципах детализации (пошаговости) и тьюторстве.

В-пятых, акцентируется проблема поиска оптимального средства обучения иностранных абитуриентов.

Современный этап развития образования характеризуется активным развитием и внедрением новых технологий, в том числе и информационных технологий.

Использование информационных технологий в образовательном процессе делает обучение более содержательным, зрелищным, существенно повышает уровень индивидуализации обучения. Данные технологии позволяют абитуриенту (студенту) работать в своем собственном режиме.

Из вышесказанного сегодняшнее состояние исследуемой проблемы позволяет утверждать, что оптимальным средством обучения иностранных абитуриентов может являться специальный сайт по математике. Данный сайт должен включать: теоретический материал по темам (достаточно детализированный), множество примеров с пошаговым объяснением, а также различные виды контроля.

Для организации изучения теоретического материала могут быть использованы следующие виды мультимедиа курсов: видеолекция (лекция преподавателя записывается на видеопленку), мультимедиа лекция (для самостоятельной работы над лекционным материалом могут быть разработаны интерактивные компьютерные обучающие программы), традиционные аналоговые обучающие издания (электронные тексты лекций, опорные конспекты, методические пособия для изучения теоретического материала и т.д.).

Контроль уровня знаний можно осуществлять с помощью различных видов тестов.

УДК 004.6

ПРОЕКТ ИНФОРМАЦИОННОЙ СРЕДЫ УПРАВЛЕНИЯ ВУЗОМ НА ОСНОВЕ СИСТЕМЫ 1С: ПРЕДПРИЯТИЕ

Логвинович В.Я., Ходин В.М.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Козинский А.А., к.пед.н., доцент*

Автоматизация управления учебным процессом – одна из старейших кибернетических задач. Процесс обучения, с одной стороны, является предметом широкого спектра наук (педагогика, психологии, физиологии и др.). С другой – собственно обучение имеет недостаточный уровень структурированности для его формализации и представления в виде алгоритма. По указанной причине автоматизации подвергаются чаще всего лишь отдельные компоненты обучения. Примерами таких компонентов являются: предъявление учебного материала или автоматизированный контроль уровня знаний. Такие автоматизированные системы будем считать системами первого вида. К автоматизированным системам второго вида отнесем программные комплексы, применяемые непосредственно для управления учебным заведением как объектом. К системам второго рода могут быть отнесены те, которые используются для управления персоналом, финансами документооборотом и т.д. В настоящей статье представлен опыт разработки автоматизированной системы управления высшим учебным заведением, получившей название BrSU-info. Рассматриваемая система может быть отнесена к системам второго рода. Разработка такой системы ведется авторским коллективом под руководством А.А. Козинского. Приведем описание некоторых особенностей проектирования на примере подсистемы Кадры.

Подсистема «BrSU-info. Кадры» (подсистема «Кадры») предназначена для автоматизации решения стандартных задач управления данными: ввод данных, отображение на дисплее сведений о работниках в виде форм стандартного учета, сохранение данных и их чтение данных, реализация стандартных запросов к базам данных. Обращение к данным выполняется с использованием технологии клиент-сервер. Защита хранимых данных от несанкционированного доступа обеспечивается средствами криптографии. Такая защита осуществляется на уровнях: авторизации пользователей, хранения данных, защиты управляющего кода системы управления базой данных. Условиями функционирования программной подсистемы «BrSU-info. КАДРЫ» является сетевая система, состоящая из сервера, работающего под управлением операционной системы и клиентов, с установленным программным обеспечением. Перечисленные особенности являются стандартными для автоматизированных систем управления.

Интерес, по мнению авторов, представляют цели проектирования. Так, основной целью создания подсистемы Кадры системы управления вузом «BrSU-info» является автоматизация управления персоналом. Дополнительной целью проектирования (и не менее важной для учебного заведения) является актуализация знаний и умений студентов в условиях, предоставляемых вузом. Последняя цель в условиях Брестского государственного университета имеет особую значимость, так как позволяет внедрять активные формы обучения студентов в условиях промышленного производства программных изделий. Такая активизация обучения важна для вузов, занимающихся подготовкой специалистов по IT-технологиям и не имеющих лабораторий-филиалов разработчиков программных изделий.

На этапе предпроектного обследования выполнено изучение предметной области. Выявлен ряд нормативных документов, определяющих информационную структуру проектируемой системы управления персоналом. К таким документам отнесены формы общих классификаторов учета данных (ОКУД). Например, Личный листок по учету кадров (Код по ОКУД 0221050004), Личная карточка (Код по ОКУД 0301002) и др. Необходимо отметить, что в условиях вуза управление персоналом имеет ряд дополнительных особенностей, которые также были учтены при создании системы.

Итоговым документом предпроектного проектирования явилось техническое задание на проектирование информационной подсистемы Кадры.

В результате длительного анализа возможных средств реализации проекта выбрана система разработки прикладных решений 1С: Бухгалтерия. Причины выбора объясняются следующими ее возможностями: наличие встроенных форм для организации хранения информации, встроенный интерфейс импорта (экспорта) данных в наиболее распространенные форматы (см. [1], [2], [3]). Важной особенностью среды разработки является наличие встроенного языка программирования, что позволяет широко использовать не только стандартные формы, но и реализовывать собственные задачи обработки данных. Немаловажная причина выбора среды 1С: Бухгалтерия – это возможность реализации международных стандартов управления предприятием.

Наибольший интерес, с точки зрения авторов, представляет этап технического и рабочего проектирования.

Внимание разработчиков на этапе проектирования привлек встроенный язык программирования. По нашему мнению, присутствие такого языка в среде разработки программных решений 1С: Бухгалтерия (1С: Предприятие 8.1) дает возможность использовать не только готовые шаблоны, что значительно ускоряет разработку программных изделий. Встроенный язык значительно расширяет возможности разработчика. Для примера использованных возможностей встроенного языка приведем программный код вычисления возможного стажа работника:

```
Фрагмент 1. Программный код вычисления возможного стажа работника
Процедура ТрудоваяДеятельностьПриемНаРаботуПриИзменении(Элемент)
    строкаТч=ЭлементыФормы.ТрудоваяДеятельность.ТекущиеДанные;
    Если строкаТч.ПриемНаРаботу= Дата(1,1,1) Тогда
        строкаТч.СтажПоСпециальности=0;
    иначе
        д1= строкаТч.ПриемНаРаботу;
        Если строкаТч.ДатаУвольнения= Дата(1,1,1) Тогда
            Д2=началоДня(ТЕкущаяДата());иначе
                Д2=строкаТч.ДатаУвольнения;
        КонецЕсли;
```

```

Лет=0;
Пока д1<д2 Цикл
    Лет=Лет+1;
    д1= ДобавитьМесяц(д1,12);
КонецЦикла;
Если Месяц(д2)< Месяц(д1) Тогда
    Лет=Лет-1;
КонецЕсли;
Если Лет<0 Тогда Лет=0 КонецЕсли;
строкаТч.СтажПоСпециальности=Лет;
КонецЕсли;
КолЛет=0;
Для каждого строчка из ТрудоваяДеятельность Цикл
    КолЛет=КолЛет+строчка.СтажПоСпециальности;
КонецЦикла;
Общий_Стаж_По_Специальности=КолЛет;
КонецПроцедуры

```

В настоящее время подсистема «BrSU-info. Кадры», которая будет продемонстрирована в ходе доклада, находится на этапе опытной эксплуатации.

Литература

1. Попов, С.А. Delphi и 1С:Предприятие. Программирование информационного обмена / С.А. Попов. — СПб. : БХВ-Петербург, 2007. — 592 с. : ил. + CD-ROM — (Профессиональное программирование).
2. Кашаев, С. М. 1С:Предприятие 8. Учимся программировать на примерах. / С.М. Кашаев. – СПб.: БХВ-Петербург, 2008. — 336 с.: ил. + CD-ROM.
3. Гончаров, Д. Введение в конфигурирование в примерах “1С:Предприятие 8”. Курс дистанционного обучения / Д. Гончаров. ООО «1С. Учебный центр №3», август 2008 г. // E-mail: Uc3@1c.ru

УДК 517

ИНТЕРАКТИВНЫЕ АНИМАЦИОННЫЕ ПРОГРАММНЫЕ КОМПЛЕКСЫ В СИСТЕМЕ ТЕХНИЧЕСКОГО ОБРАЗОВАНИЯ

Мацкевич С. В., Савицкий Ю.В.

УО «Брестский государственный технический университет», г. Брест

Одним из приоритетных направлений в области повышения качества обучения техническим дисциплинам является разработка и внедрение инновационных образовательных технологий, основанных на применении современных аппаратно-программных средств вычислительной техники. Практика применения компьютерных обучающих систем совместно с традиционными средствами обучения демонстрирует существенное улучшение качества знаний и навыков слушателей. При этом на первый план выходит задача принципиально нового построения содержания учебного материала, деятельности преподавателя и учебной работы студента в компьютерной среде.

В контексте данной задачи наиболее перспективным направлением можно считать использование *интерактивных анимационных программ (ИАП)* и элементов виртуальной реальности. Они позволяют слушателю самостоятельно исследовать процессы, протекающие в изучаемом объекте, понять основные закономерности, получить всестороннее представление об излагаемом материале. Часто компьютерная анимация позволяет наглядно представить материал, словесное описание которого объемно и достаточно сложно для восприятия, а показ на натурном образце невозможен или затруднен (например, физические, технологические, информационные процессы, функционирование адаптивных систем управления и др.). Важным положительным свойством ИАП является возможность визуализации функционирования таких объектов и систем, в которых работа различных взаимоувязанных в систему компонентов полностью или частично совмещена во времени. Большим достоинством ИАП также является возможность имитации и моделирования протекания различных явлений и процессов в реальном, ускоренном или замедленном масштабах времени; это позволяет акцентировать внимание обучаемого на принципиальных моментах функционирования системы. ИАП предлагают обучаемому не «прочтение» с помощью компьютера целого курса или его фрагментов, а более высокий уровень представления в учебном процессе самого осваиваемого объекта.

Таким образом, по мнению авторов, ИАП являются одним из важнейших компонентов современных электронных учебно-методических материалов, а процесс их проектирования и разработки, как правило, один из самых сложных в практике создания электронно-обучающих систем.

Опыт авторов в области разработки технических ИАП позволил сформулировать следующие ключевые этапы, позволяющие осуществить системный подход к проектированию и разработке анимационных средств обучения:

1. *Детальный анализ объекта анимации – предполагает всестороннее исследование натуральных образцов (моделей) и их характеристик (методов, алгоритмов, чертежей, схем функционирования объекта), технической документации, фотографий, видеоматериалов и других информационных источников.*

2. *Выделение принципиальных (ключевых) событий функционирования объекта – имеет своей целью обобщить информацию об объекте и определить перечень сцен, которые, с научно-методической точки зрения, в наибольшей степени отражают специфику работы исследуемого объекта. На данном этапе также следует определить степень детализации и формализации отдельных фрагментов анимируемых процессов.*

3. *Определение структуры динамических сцен в ИАП и построение графа переходов.*

4. *Предварительная разработка сценариев динамических иллюстраций. Реализация данного этапа предполагает: детализацию сцен до уровня отдельных компонентов; проектирование интерфейса взаимодействия с пользователем; определение расположения основных и периферийных объектов в окне сцены; решение вопросов изображения крупным планом принципиальных компонентов сцен и схематизации иных визуальных объектов, наличие которых в сцене необходимо для понимания слушателем принципа функционирования системы в целом.*

5. *Проектирование и разработка отдельных базовых элементов динамических сцен (конкретных объектов, систем, подсистем и др.) – осуществляется, как правило, с использованием программных систем векторной графики [1-3]; при этом положительным свойством анимационных инструментов является возможность импортирования готовых компонентов. Важным аспектом разработки на данном этапе является ведение и использование библиотек базовых элементов, что во многих ситуациях позволяет унифицировать и существенно снизить трудоемкость программирования сцен.*

6. Программирование видеоматериалов сцен, включающее в себя разработку опорных (ключевых) кадров и анимационных переходов между ними.

7. Программирование обработчиков элементов управления сценами с использованием встроенного языка анимационного редактора.

8. Обязательная разработка подсистемы методических указаний, включающих детальные описания характеристик и принципов работы исследуемого объекта, элементов управления анимацией и правил их использования для демонстрации возможных режимов работы объекта. Отсутствие в программе информации подобного типа в ряде случаев превращает анимацию в «головоломку», снижая практически до нуля обучающий эффект от разработки.

9. Согласование сцен и интегрирование в единую систему.

10. Обязательное тестирование полученного варианта системы с привлечением в качестве экспертов лиц соответствующей квалификации – необходимо, во-первых, для отслеживания ошибок, неизбежно возникающих при создании программных средств, а во-вторых, для проверки адекватности анимированных процессов реальным, имеющим место в существующем техническом объекте.

11. Опубликование, защита и электронное тиражирование ИАП.

Разработанные авторами принципы построения ИАП были практически реализованы в интерактивных обучающих программных системах: «Маршрутизация потоков в базовой сети обмена данными» и «Принципы межсетевого взаимодействия по протоколу без установления соединения Internet Protocol (IP)».

Данные системы представляют собой программные комплексы для исследования принципов межсетевого взаимодействия и адаптивной маршрутизации в объединенных гетерогенных IP-сетях; позволяют на базе современных средств информационных технологий обеспечить всестороннее изучение наиболее сложных динамически протекающих многоуровневых процессов передачи и маршрутизации в вычислительных сетях TCP/IP [4] с активным участием обучаемого. При осуществлении указанных процедур используется совокупность последовательно и параллельно функционирующих процессов и протоколов, реализующих распределенные алгоритмы на абонентских системах, сетевых шлюзах (маршрутизаторах) и др. Очевидно, что такая специфика материала (при его представлении в традиционной форме) создает объективные затруднения в его комплексном понимании.

Вышеуказанные программные средства разработаны в рамках научно-исследовательской работы ЭИ-08/06 «Современные интеллектуальные технологии обработки информации», ГР № 2008553; внедрены в учебный процесс кафедры «Интеллектуальные информационные технологии» в 2009 году.

Таким образом, современные информационные технологии предлагают сегодня широкие возможности для создания высокоэффективных компьютерных средств, позволяющих внедрять в образовательный процесс активные методики обучения. Однако важно понимать, что наилучший эффект может быть достигнут только при использовании компьютерных разработок в совокупности с традиционными, проверенными временем и практикой, методами обучения.

Литература

1. Жданов, А. Flash 5. Краткий курс / А. Жданов – СПб.: Издательский дом "Питер", 2001. – 324 с.

2. Аврамова, О.Д. Язык VRML: практическое руководство / О.Д. Аврамова – М.: Диалог-МИФИ, 2001. – 288 с. – ISBN 5-86404-154-8

3. Матоссян, М. 3DS MAX для Windows / М. Матоссян – М.: ДМК Пресс, 2004. – 624 с.

4. Олифер, В.Г. Компьютерные сети. Принципы, технологии, протоколы: учебник для вузов, 3-е издание / В.Г. Олифер, Н.А. Олифер – СПб: Издательство «Питер», 2007.

УДК 517

ПРОБЛЕМЫ ПРОЕКТИРОВАНИЯ СТРУКТУРИРОВАННЫХ КАБЕЛЬНЫХ СИСТЕМ ЛОКАЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ

Мельничук А.В., Полуянович Л.В., Савицкий Ю.В.

УО «Брестский государственный технический университет», г. Брест

В настоящее время все больше сфер человеческой деятельности могут быть автоматизированы посредством информационных технологий (ИТ). Фундаментом внедрения ИТ-проектов, позволяющим в дальнейшем использовать все заложенные в них возможности, является *структурированная кабельная система* (СКС) [1]. Проектирование СКС, ориентированной на специфику решаемых предприятием задач, позволяет уменьшить затраты на создание СКС и повысить эффективность дальнейшей работы системы в целом. Это особенно важно, так как из всех составляющих ИТ-проекта СКС имеет наименьшую стоимость внедрения и наибольший срок эксплуатации. Учет особенностей деятельности предприятия, реализованный еще на этапе проектирования, позволит получать отдачу от вложений в СКС в течение всего срока ее эксплуатации. В то же время ошибки, допущенные при проектировании, в дальнейшем могут привести к капитальной реконструкции СКС.

В качестве возможных приложений могут выступать распределенные сети передачи голоса, данных, мультимедийного трафика, системы передачи служебных сигналов [2]. В последнее время в результате развития IP-технологий проявляется тенденция объединения систем в *мультисервисные и конвергентные сети*. Количество и разнообразие интегрируемых в конвергентные сети систем диктуется наличием оборудования, поддерживающего цифровые стандарты. Для проектирования СКС разработаны стандарты TIA/EIA-568 Commercial Building Telecommunications Cabling Standard и TIA/EIA-569 Commercial Building Standard for Telecommunications Pathways and Spaces TIA/EIA-570 Residential Telecommunications Cabling Standard, а также их международные и европейские аналоги.

В настоящее время профильные нормативные документы и рекомендации предоставляют все возможности для реализации кабельной инфраструктуры любой сложности. Задача проектировщика состоит в определении топологии, параметров системы и ее технических характеристик, а также в корректном понимании требований, предъявляемых к СКС, которая применяется к конкретному объекту. При этом базисными принципами построения СКС являются: *реконфигурируемость, надежность, гибкость, избыточность, масштабируемость, приемлемая стоимость внедрения и эксплуатации* [1,2].

Перечень проектной документации может варьироваться и, как правило, включает в себя общие данные, структурную схему, схему расположения оборудования и кабельных трасс, схему расположения кабельных каналов, схему электрических соединений, кабельный журнал и спецификацию материалов и оборудования.

Опишем порядок действий при проектировании СКС на объектах в виде алгоритма.

1. Обозначить функциональные зоны, которые существуют в здании.
2. Выделить возможные приложения, сервисы, типы подключаемого оборудования.
3. Определить требования, предъявляемые к СКС, и возможности по ее реализации.
4. Составить топологию системы, определить ее технические характеристики и требования по исполнению.

5. Рассмотреть возможности по расположению оборудования, трасс кабелей и кабель-каналов в функциональных зонах.

Вместе с тем, при реализации поставленных задач существует ряд принципиальных вопросов, от эффективности решения которых зависят как эксплуатационные, так и стоимостные характеристики СКС [3-5]. При этом открытые информационные источники в области проектирования СКС, по мнению авторов, практически не отражают методы и алгоритмы решения данных проблем. Применительно к СКС локальных вычислительных сетей (ЛВС) можно выделить следующие вопросы, требующие системного подхода к решению:

1. Анализ внешнего трафика, зависящего от: назначения СКС; типа программных приложений; характеристик прикладных задач, реализация которых возлагается на сеть информационного обмена; характера и мощности информационных потоков между сетевыми абонентами.

2. Аналитический расчет пропускных способностей каналов СКС, а также задержек передачи данных; на основании этого (при необходимости) – уточнение топологии СКС с учетом требований и ограничений соответствующих стандартов на ЛВС.

3. Выбор каналов с соответствующими характеристиками.

4. Выбор активного/пассивного сетевого оборудования ЛВС с характеристиками, увязанными с вышеизложенными расчетами базовых параметров СКС.

Литература

1. Семенов, А.Б. Проектирование и расчет структурированных кабельных систем и их компонентов / А.Б. Семенов – ДМК-Пресс, 2008. – 416 с.

2. Семенов, А.Б. Волоконно-оптические подсистемы современных СКС / А.Б. Семенов – ДМК-Пресс, 2008. – 632 с.

3. Олифер, В.Г. Компьютерные сети. Принципы, технологии, протоколы: учебник / В.Г. Олифер, Н.А. Олифер – СПб.: Питер, 2001. – 672с.

4. Стерлинг, Д.Дж. Кабельные системы / Д.Дж. Стерлинг, Л. Бакстер – Лори, 2003. – 316 с.

5. Бердсекас, Д. Сети передачи данных: пер. с англ / Д. Бердсекас, Р. Галлагер – М.: Мир, 1989. – 544с.

УДК 004.514.62

ПРИМЕНЕНИЕ БЕСКЛИКОВОЙ НАВИГАЦИИ В ГРАФИЧЕСКОМ ИНТЕРФЕЙСЕ ПОЛЬЗОВАТЕЛЯ

Морозенко Г.П., Сидорович А.С.

УО «Брестский государственный технический университет», г. Брест

С тех пор, как указательные устройства ввода стали широко внедряться в компьютерную технику, предоставив альтернативу обычной клавиатуре, пользователям персональных ЭВМ сразу же стали очевидны все положительные стороны перспективы их использования. Наибольшее распространение получил манипулятор «мышь», который до сих пор для многих людей ассоциируется с понятием «компьютер». Введение «указательной» концепции графического интерфейса предоставило разработчиками программного обеспечения широкий спектр возможностей по облегчению взаимодействия с компьютером, которые не могли были быть раньше реализованы.

Наряду с введением мыши (и иных указательных устройств ввода, таких как трекбол, тачпад, трекпоинт) в сознании пользователей персональных компьютеров крепко зафиксировалось понятие «клик», обозначающее нажатие кнопки на манипуляторе. Наличие не одной, а нескольких кнопок в значительной степени расширяет диапазон возможных действий, осуществимых в результате комбинирования кликов с нажатием клавиш на стандартной клавиатуре. «Клик» стал олицетворением подтверждающего действия пользователя. В то же время нельзя утверждать, что все программные средства, выпущенные в различное время, были ориентированы строго на работу пользователя исключительно кликами. Уже давно в разных средах находит себе применение бескликовая навигация, основанная на выполнении того или иного действия не при щелчке на наведенном объекте, а путем наведения курсора на объект. Можно привести несколько всем известных примеров стандартных интерфейсных решений, где пользователь сталкивается с бескликовыми действиями:

- всплывающие подсказки, закрепленные за определенными кнопками и пиктограммами;
- выпадающие меню, появляющиеся на экране при наведении курсора на соответствующую пиктограмму;
- подсветка кнопок и гиперссылок (часто используется вместе с всплывающими подсказками).

Данные примеры нельзя однозначно назвать бескликовой навигацией, которая подразумевает полный отказ от кликов, однако приведенные действия реализуют дополнительные возможности в классических интерфейсах, основанных на работе с кликами.

Принцип функционирования полностью бескликового интерфейса заключается в том, что все действия, которые только можно выполнить, производятся исключительно наведением указателя на объект и удерживания его в этом положении в течение фиксированного отрезка времени (теоретически продолжительность может стремиться к нулю, обеспечивая эффект мгновенного срабатывания).

Следует учитывать, что применение бескликового интерфейса несёт в себе как положительные, так и отрицательные моменты.

Недостатки бескликовой навигации:

- Неудобство в использовании для людей, привыкших работать с кликами. Объясняется тем, что для стандартного пользователя клик по объекту является настолько привычной процедурой, что переучиться он сможет лишь спустя некоторое время. И не факт, что пользователь пожелает отказываться от прежних привычек.
- Сложность управления – работа с таким интерфейсом требует от пользователя некоторых навыков, так как необходимо иметь как минимум четкую координацию движений. Иначе, в случае недостаточной координации, неизменным спутником такой работы будут частые ошибки, возникающие из-за неточного наведения курсора на нужный в конкретный момент объект.
- Сложность грамотной реализации интерфейса. Бескликовый интерфейс невозможно создать путем простой замены всех кликовых операций на бескликовые, так как недостаточная проработка интерфейса повлечет за собой еще большее число ошибок со стороны пользователя — ошибок, которые будут вызваны не просто человеческим фактором, а физической невозможностью работать эффективно.

- Принудительный отказ от многих возможностей, предоставляемых обычным кликовым интерфейсом, либо же поиск их замены.

К несомненным достоинствам бескликового интерфейса, в свою очередь, можно отнести:

- Применение бескликовых операций при грамотной организации рабочего процесса способно увеличить производительность работы за счет постоянного движения и в целом нахождения рабочей среды в динамическом состоянии, поддерживающем у пользователя соответствующий тонус.

- За счет увеличения скорости работы возможно достичь более быстрых способов доступа к информации.

- Работа с программами, организованными по бескликовому принципу, значительно лучше сосредотачивает пользователя ввиду того, что увеличен риск ошибки, и пользователь не может себе позволить быть невнимательным.

Как видно из сказанного, все положительные моменты бескликового интерфейса достижимы лишь при выполнении двух условий:

- профессионально реализованная рабочая среда, обеспечивающая максимально возможное удобство в обращении с собой;

- грамотно подобранные значения фиксированных длительностей времени нахождения указателя на объекте, предшествующих срабатыванию действий, закрепленных за объектами.

Управляя этими двумя показателями (отчасти только вторым, так как первое условие должно быть выполнено уже на этапе проектирования программного обеспечения), можно достичь положительного эффекта. В противном случае вряд ли можно ожидать действительной пользы от такого вида интерфейса.

Нельзя утверждать, что в настоящее время не существует программных средств, разработчики которых отказались от стандартных кликов, заменив их полностью на бескликовые эквиваленты. Рассмотрим несколько примеров:

Web-сайты. На данный момент известно некоторое количество сайтов, созданных полностью на основе бескликового интерфейса. Пока что это далеко не глобальные порталы, а всего лишь сайты, несущие ознакомительную информацию.

Благодаря своей оригинальности, такие web-страницы способны заставить пользователя обратить на себя внимание и превратить работу с собой в своеобразную игру.

Самым удачным web-проектом в данной сфере многими пользователями признан сайт www.dontclick.it [1], который не только создан с применением бескликовых технологий, но и сам по себе предназначен для популяризации концепции отказа от привычных кликов.

Разработчики сайта предлагают на суд посетителей не только широкий спектр возможностей бескликового интерфейса, но и вводят в обиход новые понятия. К примеру, на сайте можно встретить различные варианты условия срабатывания действия, привязанного к кнопке:

- кнопка считается нажатой в случае, если пользователь провел указателем мыши по кнопке в определенном направлении, например, слева направо или наоборот (рис. 1-а).

- эквивалентом клику признается совершение некой манипуляции над кнопкой, например, очерчивание вокруг нее условной окружности (рис. 1-б).

- клик фиксируется по истечении времени пребывания курсора на кнопке (рис. 1-в).

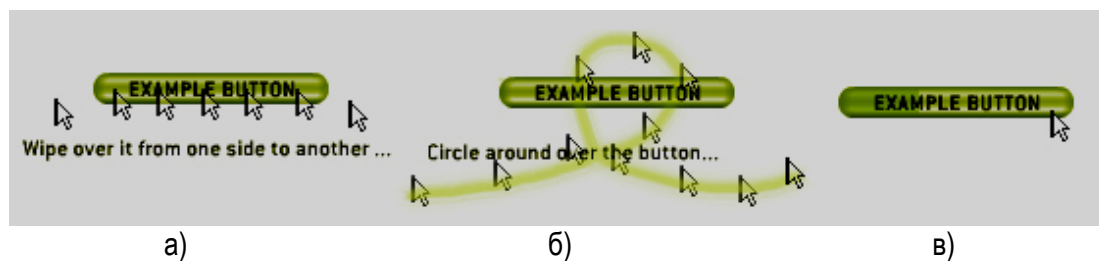


Рисунок 1 – Демонстрация вариантов замены клика

Также в качестве эффективного способа активации бескликовой кнопки можно отметить вариант, представленный на рис. 2. При наведении курсора на подлежащую активации кнопку, на экране появляется дополнительная кнопка, являющаяся логическим продолжением данной; пользователь должен перевести на нее курсор, подтверждая тем самым свой выбор.

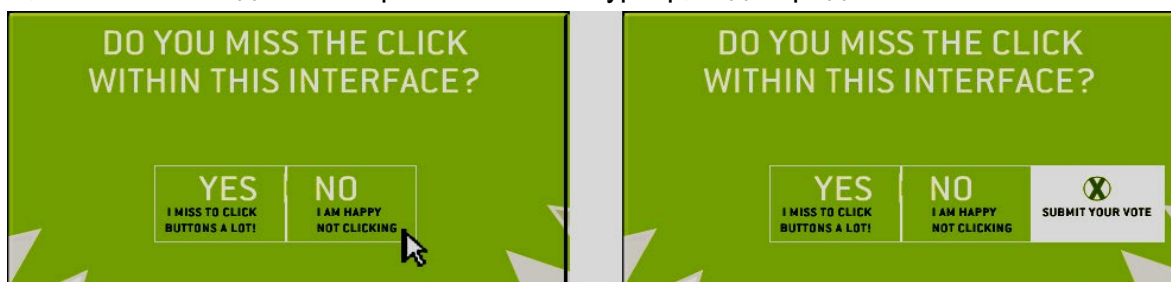
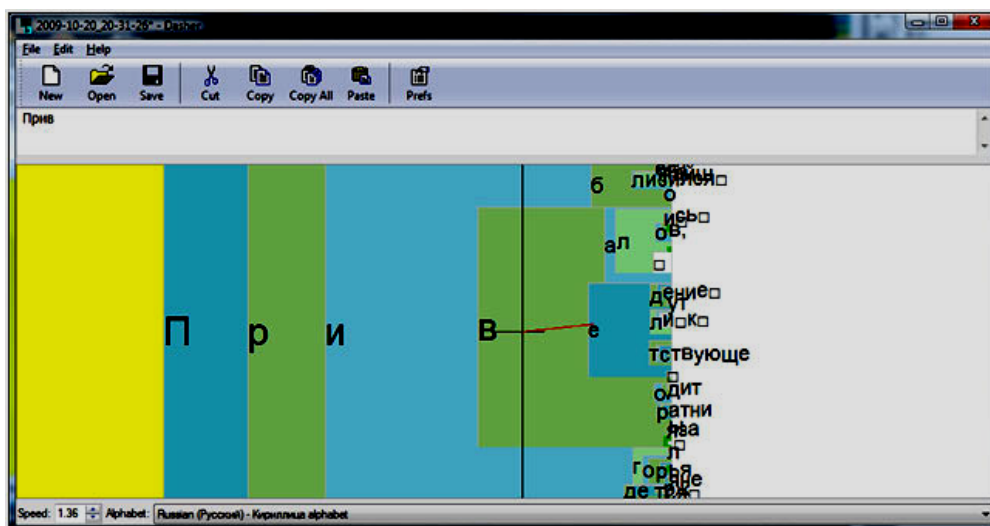


Рисунок 2 – Демонстрация всплывающего меню

Программы, облегчающие работу людям с ограниченными способностями. Отсутствие кликов в интерфейсе может быть и вынужденной мерой. Речь идет о людях, для которых (ввиду определенных обстоятельств) управление персональным компьютером в привычном понимании этого слова представляет сложную задачу. Наиболее известным примером является программа **Dasher** [2], предназначенная для предиктивного набора текста. Она позволяет пользователям вводить текст без использования клавиатуры, при помощи любого указательного устройства.

Принцип действия программы основан на том, что пользователю не нужно совершать никаких других действий, кроме движения манипулятора по направлению к букве, кото-



рая ему необходима. По достижении выбранной буквы открываются на выбор символы, из которых он выберет очередную букву, и так далее.

Рисунок 3 – Рабочее окно программы Dasher

Литература

1. В «Рунет» пришла мода на бескликовые сайты. 18/11/2008. http://habrahabr.ru/blogs/web_design/44858/
2. Inference Group: Dasher Project: What is Dasher for. 19/10/2009. <http://www.inference.phy.cam.ac.uk/dasher/DasherSummary.html>

УДК 681.3

ОЦЕНКА ИТ-ПРОЕКТОВ

Мулькевич Н.Н., Хвещук В.И.

УО «Брестский государственный технический университет», г. Брест

Актуальность проблемы. Оценка ИТ-проекта - один из важных и ключевых этапов разработки систем обработки данных (СОД). СОД как объект оценки состоит из таких компонентов, как программное, информационное и техническое обеспечение. Программное обеспечение (ПО) СОД является определяющим в стоимости системы. Для оценки ИТ-проектов используют такие параметры, как людские ресурсы, продолжительность создания (разработки) СОД и её стоимость. Оценка проекта выполняется клиентами, поставщиками решений, и инвесторами. В качестве основных целей оценки проектов могут быть: принятие решения о целесообразности проекта; сравнение вариантов автоматизации в процессе выбора; проведение переговоров о стоимости проекта; планирование расходов на проект (бюджетирование); контроль фактических расходов на проект. Среди факторов, влияющих на неправильную оценку разработки, можно выделить следующие: незнание методик оценки проекта или отсутствие опыта; неправильная оценка рисков проекта; ошибка аналитиков в оценке трудоёмкости; недопонимание ключевых технических проблем проекта; недостаток времени на изучение документации заказчика ПО и другие.

Модели, методы и средства оценки ИТ-проектов. Можно выделить следующие основные методы оценки проектов [5]: аналогий, аппроксимации, директивный и затратный.

В оценке стоимости ПО используют две единицы размера:

1. Строка исходного кода ПО (LOC – Line of Code), за исключением пустых строк, комментариев и специфических операторов. LOC-оценки относятся к группе размерно-ориентировочных метрик, которые прямо измеряют программный продукт и процесс его разработки.

2. Функциональная точка (FP – Function Point). Функциональные точки (FP) относятся к группе функционально-ориентировочных метрик, которые косвенно измеряют программный продукт и процесс его разработки. При этом рассматривается не размер, а функциональность или полезность продукта. Также важно отметить, что FP-оценки легко пересчитываются в LOC-оценки.

Данные единицы оценки размера ПО, базируются на метриках – мерах, позволяющих получить численное значение некоторого свойства программного обеспечения или его спецификаций.

Основным методом, измеряющим функционально-ориентировочные метрики, является метод функциональных точек, а также производные от него методы (точки свойств, Mark II, трехмерные функциональные точки).

Функциональные точки [2]. Методика анализа FP основывается на концепции разграничения взаимодействия. Сущность ее состоит в том, что программа разделяется на классы компонентов по формату и типу логических операций. В основе этого деления лежит предположение, что область взаимодействия программы разделяется на внутреннюю – взаимодействие компонентов приложения, и внешнюю – взаимодействие с другими приложениями.

Точки свойств (Feature Points). Используется в условиях, когда сформулированные требования не отражают истинной сложности реализации (что особенно характерно для системного ПО, критически важных программных комплексов и пр.). Метод точек свойств учитывает не только требования к системе, но и внутренние особенности ее реализации. В отличие от метода функциональных точек, он предусматривает корректирование получаемой оценки с учетом алгоритмической сложности. Эта методика считается экспериментальной.

Метод Mark II [4]. Это модификация метода функциональных точек, которая позволяет избавиться от многих известных его недостатков и сделать более пригодным для оценки сложных систем. В частности, Mark II позволяет добиться одного и того же результата как при оценке системы в целом, так и при суммировании оценок, полученных для составляющих ее подсистем.

Трехмерные функциональные точки [4]. Еще одно логическое развитие оригинального подхода метода функциональных точек было предложено корпорацией Boing. В основу этого метода положена идея о том, что сложность задачи в программной среде можно представить в трех измерениях – данные (количество вводов/выводов), функции (сложность вычислений) и контроль (управляющая логика). Важно отметить, что он выходит за рамки исключительно программных проектов и позволяет оценивать трудоемкость решения задач в различных сферах – деловой, научной и т. д.

Также широко распространены методы оценки проектов с использованием эмпирических данных, а именно: Wideband Delphi, Метод ДеМарко, SLIM, COCOMO, COCOMO II. Данные методы используют как размерно-ориентировочные, так функционально-ориентировочные метрики.

Wideband Delphi [4]. Широко используемая в классическом менеджменте технология экспертной оценки по методу Delphi, применяется при оценке программных проектов.

Метод ДеМарко [4]. Относительно простой, но эффективный подход к оценке стоимости ПО на основе накопленного опыта. Основан на использовании так называемой «бэнг-метрики», близкой по своему содержанию к функциональным точкам. Главная особенность состоит в том, что оценки корректируются с учетом хронологических данных по выполненным ранее проектам, что позволяет получить не абстрактные показатели, а приближенные значения реальных затрат ресурсов и времени. Последовательное и систематическое применение данного метода позволяет постепенно повышать точность оценок.

COCOMO (COConstructive COst MOdel) [3,4] – конструктивная модель стоимости. Создана на основе анализа статистических данных 63 проектов различных типов. Фактически под общим названием скрываются три уровня детализации: базовый, промежуточный и подробный.

- базовая (базисная) COCOMO – статическая модель, вычисляет затраты разработки и её стоимость как функцию размера программы;
- промежуточная COCOMO – дополнительно учитывает атрибуты стоимости, включающие основные оценки продукта, аппаратуры, персонала и проектной среды;
- подробная (усовершенствованная) COCOMO – объединяет все характеристики промежуточной модели, дополнительно учитывает влияние всех атрибутов стоимости на каждый этап процесса разработки ПО (анализ, проектирование, кодирование, тестирование и т.д.).

Предусмотрены три режима использования модели в зависимости от размеров команды и проекта: ограниченный, заблокированный и внедренный.

Основа СОСОМО – модель, которая вычисляет стоимость разработки ПО в зависимости от оценок размера кода программы и комплекса "издержек", которые включают субъективную оценку товара, оборудования, персонала и проектных характеристик. Есть различные варианты модели СОСОМО, которые включают все характеристики, с оценкой стоимости управляющих воздействий на каждый шаг (анализ, проектирование и т.д.) в процессе разработки программного обеспечения. Модель вводит 15 поправочных факторов, принадлежащих к одной из четырех категорий, которые в свою очередь получают оценку по 6 – балльной шкале: атрибуты продукта, атрибуты системы, атрибуты команды разработчиков и атрибуты проекта.

СОСОМО II [1,4]. Она усовершенствует оригинальную модель в следующих основных направлениях:

- использование входных данных, доступных на ранних этапах жизненного цикла системы для оценки ее сложности (в частности, использование функциональных точек);
- подходы, основанные на повторном использовании, включая интеграцию коммерческих продуктов, реинжиниринг, генерацию приложений;
- объектно-ориентированные подходы, поддерживаемые распределенным ПО промежуточного слоя;
- влияние зрелости процессов разработки;
- новые – циклические и обобщенные – модели процессов разработки.

При построении СОСОМО II для обработки статистических данных использовался Байесовский анализ, который дает лучшие результаты для программных проектов, характеризующихся неполнотой и неоднозначностью, в отличие от многофакторного регрессионного, примененного в СОСОМО. Допускается измерение размера проекта не только числом строк кода, но и более современными функциональными и объектными точками. СОСОМО II также имеет несколько вариантов использования: композиционная прикладная, ранней разработки проекта, постархитектурная модель.

На мировом рынке средств автоматизации оценки ИТ-проектов можно выделить следующие разработки:

- Microsoft Project – средство управления проектами (Microsoft);
- Wideband Delphi - технология экспертной оценки по методу Delphi (Rand Corporation);
- SLIM Estimate – средство оценки стоимости разрабатываемого ПО, основанное на модели SLIM (QSM).
- Costar – средство оценки стоимости ПО, основанное на модели СОСОМО (SoftStar Systems).

Все приведенные выше средства являются корпоративными разработками для конкретных задач и не дают необходимой масштабируемости и универсальности при оценке ИТ-проектов.

Постановка задачи на создание системы. В результате обзора и анализа методов, подходов и средств предложен проект системы «Оценка ИТ-проектов», ориентированный на автоматизацию получения оценок затрат на различных этапах (фазах) процесса создания СОД с использованием различных методов и подходов. В рамках данной системы оценка делается исходя из себестоимости отдельных составляющих проекта, для которой будет использоваться затратный метод. На каждом этапе создания проекта бу-

дуг оцениваться основные технико-экономические показатели (ТЭП). Для их расчета будут использованы три методики: экспертные оценки ТЭП проектов СОД при подготовке концепции и технического задания; ТЭП проектов СОД с учетом совокупности основных факторов предварительной модели СОСОМО II; уточненная оценка ТЭП проектов СОД с учетом полной совокупности факторов детальной модели СОСОМО II.2000. Реализация системы предполагается в рамках дипломного проекта.

Литература

1. Липаев, В.В. Программная инженерия: Методологические основы / В.В. Липаев – Москва, 2006.
2. Орлов, С.А. Технология разработки программного обеспечения / С.А. Орлов – СПб.: Изд. дом «Питер», 2002.
3. Электронный научный журнал «Исследование в России», <http://zhurnal.ape.relarn.ru/articles/2008/030.pdf>
4. Колдовский, В. Разработка ПО: оценка результата, 21 сентября 2006г., <http://itc.ua/node/25631>
5. Колтунова, Е. Как оценить стоимость проекта автоматизации?, <http://www.koltunova.com/it-project-costs/>

УДК 004:336.71(476)

АВТОМАТИЗИРОВАННЫЕ БАНКОВСКИЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ ЗАО «СОФТКЛУБ»

Ничипорук Т.С.

УО «Полесский государственный университет», г.Пинск

Динамично изменяющиеся условия рынка банковских услуг постоянно требуют адекватной реакции от банковских аналитиков. От того, насколько технологически эффективно и гибко организован процесс решения аналитической задачи, во многом зависит оперативность получения результатов и их качество.

Центром банковских технологий разработано и обеспечивается методическое сопровождение Концепции развития и применения информационных технологий в банковской системе Республики Беларусь на 2008–2012 гг. Концепция направлена на развитие дистанционных форм обслуживания потребителей банковских услуг и интеграцию единого информационного пространства банковской системы с информационным пространством государственных органов управления. В соответствии с Концепцией также предусмотрены разработка и создание централизованной автоматизированной информационной системы ведения нормативно–справочной информации, автоматизированной системы управления банковскими рисками, создание единой интеграционной платформы на базе системы гарантированной доставки юридически значимых электронных документов, создание структурированного резервного центра.

Крупнейшим центром разработки программных продуктов в Республике Беларусь является ЗАО «СОФТКЛУБ – Центр разработки». Главным направлением его деятельности является разработка программных технологий и комплексных прикладных программных продуктов для автоматизации деятельности банковских и финансовых организаций, а также оказание сопутствующих услуг в отношении разработанного программного обеспечения.

Продукты «СОФТКЛУБ»:

1. Интегрированная банковская система **«SC-BANK NT»** построена на основе технологии единого информационного пространства и современной платформе СУБД ORACLE и обеспечивает автоматизацию операционной, учетной, управленческой и аналитической деятельности многофилиального банка. В основу ИБС «SC-BANK NT» положен модульный принцип построения системы, который позволяет гибко настраивать и адаптировать систему в текущую информационную среду банка и обеспечить интеграцию системы с программными комплексами других разработчиков. Таким образом, обеспечивается комплексная автоматизация всех основных бизнес-процессов банка.

2. Централизованная система **«SC-KASSA on-line»** предлагает новые подходы к автоматизации розничной банковской деятельности, позволяющие оптимизировать бизнес-процессы работы с физическими лицами в отделениях банка в соответствии с перспективными требованиями, что в итоге приводит к удешевлению себестоимости продуктов и услуг банка, повышению пропускной способности отделений, способствует повышению конкурентоспособности банка.

3. Комплексная автоматизированная система учета и обработки денежной наличности **«SC-НДО»** предназначена для управления потоками денежной наличности и ценностей, а также значительного повышения эффективности учета, обработки и степени сохранности денежной наличности и ценностей путем автоматизации технологических процессов ряда служб учреждения банка, вовлеченных в налично-денежное обращение и связанных с организацией работы по непрерывному движению наличных средств.

4. Решение **SC-АНАЛИТИКА** – это повседневный рабочий инструмент, который обеспечивает поддержку принятия управленческих решений на всех уровнях. Применимо везде, где необходимо выполнять произвольный анализ имеющихся операционных данных, анализ операционной деятельности. Это также универсальная технология, которая позволяет разрабатывать специализированные аналитические решения, в которых есть отчетность и представления для интерактивного анализа.

5. Программный комплекс **«Единый реестр клиентов»** предназначен для устранения проблемы функционирования многочисленных разрозненных систем в банке через осуществление их интеграции. Внедрение комплекса ведет к устранению дублирования информации, несогласованности данных и несвоевременности внесения изменений в различные базы данных, что позволит повысить эффективность ведения бизнеса банка.

6. Пакет прикладных программ **«SC-TRADE»** устанавливается на расчетно-кассовых местах предприятий торговли, общественного питания, ЖКХ, сферы услуг, образования, здравоохранения, а также на всех других предприятиях, где требуется учет приема оплаты товаров и услуг от населения.

7. **АБС «SC-BANK»** состоит из компокуемых по желанию потребителя программных модулей, объединенных общим меню. Модульная организация позволяет настраивать АБС «SC-BANK» в соответствии с технологией работы, принятой в конкретном банке. Потребитель сам выбирает оптимальную конфигурацию АБС «SC-BANK» согласно своим требованиям. Меню АБС «SC-BANK» может быть настроено для каждого пользователя отдельно. Все программные модули используют единую базу данных, что обеспечивает их тесную интеграцию и работу в режиме реального времени. Работа в режиме использования единой базы данных позволяют выполнять операции с непосредственным отражением на лицевых счетах таким образом, что в любой момент времени имеется реальная картина финансового состояния банка.

8. Система автоматизации работы с банковскими карточками **«SC-CARD»** предназначена для автоматизации бизнес-процессов эмиссии банковских карточек, ведения учета клиентов, карт-счетов, операций и проводок, комплексного обслуживания клиентов с использованием банковских карточек различных платежных систем и различных типов карточных продуктов, ведения учета банковских карточек и их заготовок в банке.

9. Пакет прикладных программ **«SC-CASH»** применяется для автоматизированной обработки операций по платежам населения за коммунальные и прочие услуги, для проведения валютно-обменных и прочих операций, подлежащих фискальному учету.

10. Система **«SC-KASSA»** позволяет банку проводить комплексное обслуживание физических лиц в банке: ведение вкладных счетов в национальной и иностранной валюте, обмен валют, прием коммунальных и прочих платежей от населения, работа с ценными бумагами и бланками, осуществление безналичных переводов по назначению и т.д.

11. **SC-KASSA link** - филиальная система обработки операций по розничным услугам банка, совершаемых в сети РКЦ на СУБД PervasiveSQL.

12. Система управления электронными архивами **SC-АРХИВ** предназначена для автоматизации процессов наполнения и управления архивами электронных документов, обеспечивает их надежное и долгосрочное хранение.

13. Электронный депозитарий **«Департамент фондовой автоматизации»** предназначен для автоматизации деятельности организаций — профессиональных участников рынка ценных бумаг по учету, хранению и расчетам по операциям с ценными бумагами.

14. Система **SC-КОМПЛАТ NT** предназначена для автоматизации информационного взаимодействия участников единого расчетного информационного пространства на основе централизованного хранилища данных плательщиков, требований к оплате поставщиков услуг.

15. Система **«SC-КРЕДИТНОЕ БЮРО БАНКА»** предназначена для сбора, накопления и использования информации о кредитных сделках и субъектах кредитных сделок. Внедрение кредитного бюро в банке позволит повысить качество управления кредитной задолженностью через прогнозирование поступления платежей и образования задолженности по кредитным сделкам (с применением средств Data Mining).

16. Программный продукт **«SC-СПЕЦФОРМУЛЯР-БАНК»** предназначен для регистрации и учета финансовых операций, подлежащих особому контролю, в целях взаимодействия с Департаментом финансового мониторинга Комитета государственного контроля Республики Беларусь.

17. Семейство программных продуктов **SC-УЧЕТ** позволяет автоматизировать учет персонала, учет труда и заработной платы, учет основных средств, учет нематериальных активов, складской учет, учет товарно-материальных ценностей в эксплуатации.

18. **SC-УЧЕТ ЦЕННОСТЕЙ** – автоматизация учета операций с мерными слитками, памятливыми и юбилейными монетами, драгоценными камнями, бланками строгой отчетности, депозитарными ячейками в коммерческом банке.

19. **SC-ЦМФР** – связь филиалов банка с национальной автоматизированной системой межбанковских расчетов.

Клиентами компании Софтклуб являются 25 из 31 банка, зарегистрированных в Беларуси. В 16 банках установлены АБС компании СофтКлуб, в числе самых крупных клиентов - ОАО «АСБ Беларусбанк» и ОАО «Белагропромбанк».

Система «SC-НДО» обслуживает инкассацию наличных денежных средств, платежных инструкций, драгоценных металлов, драгоценных камней и иных ценностей в 173 филиалах и отделениях в 7 крупнейших банках Республики Беларусь. «SC-УЧЕТ ЦЕННОСТЕЙ» является единственной специализированной системой в Республике Беларусь для централизованного учета операций банка со всеми видами ценностей. АБС «SC-BANK» внедрена в ОАО «АСБ Беларусбанк» в 1996 году и сегодня обеспечивает работу 144 филиалов и ЦБУ по всей Беларуси, в каждом из которых обслуживается в среднем около 35 000 счетов и совершается в день в среднем 20000 - 25000 операций. Систему «SC-CARD» используют 23 из 31 белорусских банков, эмитирующих карточки, в том числе ОАО «АСБ Беларусбанк», ОАО «Белагропромбанк», ОАО «БПС-Банк», а также Национальный банк Республики Беларусь.

Литература

1. www.softclub.by – сайт ЗАО «СОФТКЛУБ»;
2. www.nbrb.by – сайт Национального банка Республики Беларусь;
3. www.tup.km.ua - доклад «Банковские Информационные Системы», Г. Котелевец, А. Орехов.

УДК 004.6

МАТЕМАТИЧЕСКИЕ МОДЕЛИ ЗАЩИТЫ ИНФОРМАЦИОННЫХ РЕСУРСОВ МАЛЫХ ПРЕДПРИЯТИЙ

Охримчук В.В.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

В течение 2008-2009 года нами реализованы ряд программных систем управления базами данных малых предприятий.

Выполненные реализации СУБД показали, что удобно применять средства автоматизации проектирования, которые обладают следующими преимуществами: не требуют значительных временных затрат на разработку, внедрение и эксплуатацию; свободно распространяются и имеют достаточно полное описание для ведения собственных разработок; имеют средства защиты данных от несанкционированного использования.

Практика эксплуатации систем управления базами данных позволила установить, что простота их разработки с использованием имеющихся средств автоматизации обрачивается недостаточным вниманием к средствам защиты коммерческой информации от несанкционированного доступа.

Нами выполнен анализ возможных подходов к созданию модели данных, реализована такая модель для малого предприятия, разработана СУБД. Программные коды, реализующие представленную систему, включают методы их защиты от несанкционированного доступа.

В выполненных работах реализованы различные подходы, позволяющие снизить уязвимость простейших информационных систем для малых предприятий. Представим краткое описание одного из проектных решений, предназначенного для автоматизации управления малым предприятием – брачное агентство. Изучение соответствующей предметной области и проектирование выполнены в соответствии с заказом индивидуального предпринимателя Крылова Н.А., работающего на основе лицензии 02010/0338190 от 21.12.2007г., выданной Министерством внутренних дел Республики Беларусь.

Созданная нами система опирается на многопользовательские базы данных, для которой удобно применить реляционные модели. Реализация СУБД показала, что на различных этапах разработки удобно применять следующие средства автоматизации: AllFusion ERWin Data Modeler (ErWin), базовый пакет «Denwer», включающий набор дистрибутивов для разработки сайтов на локальном компьютере: Apache, PHP, MySQL, Perl и другие [1]. Такой выбор объясняется условиями функционирования малых предприятий. Их основным требованием является создание программного продукта для управления данными, размещёнными в глобальной сети, наиболее эффективным образом. Параметрами эффективности, выдвигаемыми заказчиками, являются: время создания, затраты на проектирование, создание и внедрение, эксплуатацию.

Процесс внедрения и эксплуатации указанных СУБД показал необходимость обязательного использования дополнительных средств защиты информации. Одной из основных причин защиты информации в условиях малых предприятий является недостаток квалифицированных специалистов для сопровождения программных продуктов и баз данных. Так, например, в случае использования Интернет версии, как правило, проблем не возникает, серверные услуги предоставляются специализированными предприятиями. Однако условия заказчика в подавляющем большинстве проектов предполагают эксплуатацию системы на локальном компьютере. При этом безопасности приложений путем правильной настройки сервера не всегда уделяется достаточное внимание. В описанной ситуации требуется обеспечить защиту каждого обращения к базе данных. Для примера приведем фрагмент обращения к функциям кодирования, используемым при редактировании данных об абонентах, информация которых содержится в базе данных (см. Фрагмент).

Фрагмент. Редактирование данных об абонентах.

```

if( isset($_POST['save']) ){
    $occupation = ""; // занятость
    for ($z=1; $z<=6; $z++)
        if( isset($occup[$z]) )
            $ocp[] = $occup[$z];
    if( isset($ocp) )
        $occupation = implode(' ', $ocp);

    $invalid = (isset($_POST['invalid'])) ? $_POST['invalid'] : 0 ;
    $ogranich = (isset($_POST['ogranich'])) ? $_POST['ogranich'] : 0;

    // кодируем некоторые параметры
    $_POST['adres'] = xorShipher( $_POST['adres'] , $password );
    $_POST['pasport'] = xorShipher( $_POST['pasport'] , $password );
    $_POST['dr'] = xorShipher( $_POST['dr'] , $password );
    $_POST['tel'] = xorShipher( $_POST['tel'] , $password );
    $_POST['surname'] = xorShipher( $_POST['surname'] , $password );
    $_POST['name'] = xorShipher( $_POST['name'] , $password );

    // SQL-запрос по редактированию абонента в базе данных
    $result=mysql_query("UPDATE `abonent` SET `pol` = '{$_POST['pol']}' `dr` =
'{$_POST['sr']}' , `rost` = '{$_POST['rost']}' , `ves` = '{$_POST['ves']}' , `region` =
'{$_POST['region']}' , `city` = '{$_POST['city']}' , `education` = '{$_POST['education']}' ,
`occupation` = '{$_POST['occupation']}' , `religion` = '{$_POST['religion']}' , `dom` =
'{$_POST['dom']}' , `smoking` = '{$_POST['smoking']}' , `alcohol` = '{$_POST['alcohol']}' , `sims`

```

```
= '{$_POST['sims']}' , `perezd` = '{$_POST['perezd']}' , `koldetey` = '{$_POST['koldetey']}' ,
`kid` = '{$_POST['kid']}' , `priemdetey` = '{$_POST['priemdetey']}' , `socznach` =
 '{$_POST['socznach']}' , `rassylka` = '{$_POST['rassylka']}' , `databegin` =
 '{$_POST['databegin']}' , `dataend` = '{$_POST['dataend']}' , `inet` = '{$_POST['inet']}' , `tel` =
 '{$_POST['tel']}' , `showtel` = '{$_POST['showtel']}' , `name` = '{$_POST['name']}' , `surname` =
 '{$_POST['surname']}' , `fathename` = '{$_POST['fathename']}' , `adres` = '{$_POST['adres']}' ,
 `pasport` = '{$_POST['pasport']}' , `hobbi` = '{$_POST['hobbi']}' , `invalid` = '{$_POST['invalid']}' ,
 `invalid_desc` = '{$_POST['invalid_desc']}' , `ogranich` = '{$_POST['ogranich']}' , `target` =
 '{$_POST['target']}' WHERE `id` = '{$_POST['oldid']}");
```

```
print("<center><font class=\"good\"><a href=\"index.php?do=founum&num={$id}\">Абонент
№ {$id} успешно изменён.</a></font></center><br>");
$user=getabonent($who); }
```

Приведенный выше код потребовал усовершенствования путем применения стандартных и авторских подходов.

В результате нами реализованы некоторые подходы для защиты php кода и баз данных. Используются как базовые, так и дополнительные алгоритмы шифрования. Например, хэширующий алгоритм md5, для создания цифровых подписей, позволяющих однозначно идентифицировать отправителя. В PHP для указанного алгоритма существует специальная функция. Другие подходы по защите информации баз данных малых предприятий будут продемонстрированы в ходе доклада.

Литература

1. Охримчук, В.В. Организация работы малого предприятия с использованием автоматизированной системы управления «Брачное агентство» / В.В. Охримчук // Материалы II международной студенческой конференции «Студенческая наука – будущее государства» (Пинск, 25 марта 2008 года). – Пинск: Изд-во Полеский ГУ. – 2008. – Ч.2. – С. 103.
2. Охримчук, В.В. Информационные технологии WEB-программирования базы данных управления малым предприятием // Материалы IV международной научно-практической студенческой конференции «Содружество наук. Барановичи 2008» (Барановичи, 21 мая 2008 года). – Барановичи: Изд-во РИО БарГУ. – 2008. – Ч.1. – С. 42.
3. Соколов, А. Защита от компьютерного терроризма: справочное пособие / А. Соколов, О. Степанюк. – Арлит.: БХВ-Петербург, 2002. – С. 408 – 409.

УДК 519.171.2

НЕКОТОРЫЕ СПОСОБЫ РИСОВАНИЯ ГРАФОВ НА ПЛОСКОСТИ

Райченок С.В.

УО «Брестский государственный технический университет», г. Брест

Определение проблемы. Нарисовать граф красиво - это проблема, которая постоянно возникает в приложениях, таких как отображение файла или каталога деревьев, диаграммах в схемотехнике. Алгоритм должен таким образом отражать структуру графа, чтобы зритель мог лучше понять его.

Определим несколько более жестких критериев, чем понятие «красота», оценки качества рисования графов: минимизация числа пересечений, ограничение свободного места, минимизация изломов минимизация общей длины ребер.

На основе вышеперечисленных критериев, сформулируем задачу следующим образом: описать алгоритм, переводящий матрицу смежности вершин в графовое представление, оценить степень его эффективности. Желательна максимальная адаптация алгоритма для визуализации плоских трехсвязных графов.

Решение задачи

Физические модели. В качестве алгоритмов нахождения изображения графа будем использовать алгоритмы методов физических аналогий. Были применены следующие аналогии. Будем рассматривать граф как систему тел с силами, взаимодействующих между телами. Каждая конкретная раскладка графа определяет одно из состояний этой системы. Введем понятие энергии системы, задача сводится к нахождению состояния с минимальной энергией.

Пружинный алгоритм. Граф рассматривается как система тел с силами, взаимодействующими между телами, например, считая вершины графа телами, а ребра пружинами. Задача сводится к задаче о нахождении расположения тел с локально минимальной энергией - конфигурацию равновесия сил, в которой каждое тело занимает такую позицию, что сумма всех сил равна нулю. Модернизируем алгоритм, представим, что тела, соединенные пружинами являются телами, несущими электронный заряд одинаковой величины, что позволит не приближаться вершинам слишком близко друг к другу (возможно и применение механической аналогии - связи всех вершин друг с другом дополнительными наборами пружин).

В результате, сила, действующая на тело p , определяется по формуле (1):

$$F(p) = \sum_{u=(p,q) \in E} f_u + \sum_{(p,q) \in V^2} g_{(p,q)}, \quad (1)$$

где f_u - сила растяжения, действующая на вершину p из-за пружины (p,q) , а $g_{(p,q)}$ - это сила отталкивания, существующая между частицами p и q . По закону Фуке f_u пропорциональна разности между расстоянием от p и до q и длиной пружины с минимальной энергией; а сила $g_{(p,q)}$ следует обратному квадратичному закону. Получаем формулы (2), (3) для расчета силы в двумерном (плоском случае):

$$\sum_{u=(p,q) \in E} k_u^{(1)} (d(p,q) - l_u) \frac{x_p - x_q}{d(p,q)} + \sum_{(p,q) \in V^2} \frac{k_{(p,q)}^{(2)}}{(d(p,q))^2} \frac{x_p - x_q}{d(p,q)}, \quad (2)$$

$$\sum_{u=(p,q) \in E} k_u^{(1)} (d(p,q) - l_u) \frac{y_p - y_q}{d(p,q)} + \sum_{(p,q) \in V^2} \frac{k_{(p,q)}^{(2)}}{(d(p,q))^2} \frac{y_p - y_q}{d(p,q)}, \quad (3)$$

где l_u - естественная (с нулевой энергией) длина пружины, $k_u^{(1)}$ - коэффициент жесткости пружины, а $k_{(p,q)}^{(2)}$ - коэффициент отталкивания между p и q .

Для подсчета оптимального состояния системы будем использовать градиентный метод. У градиентного метода есть существенный недостаток – метод может свалиться в локальный экстремум, не достигнув глобального экстремума. Оптимизируем алгоритм, добавив в него силу притяжения к центру области, на которой происходит рисование. Так же первоначальное положение системы будем выбирать не случайно, а располагать вершины по кругу.

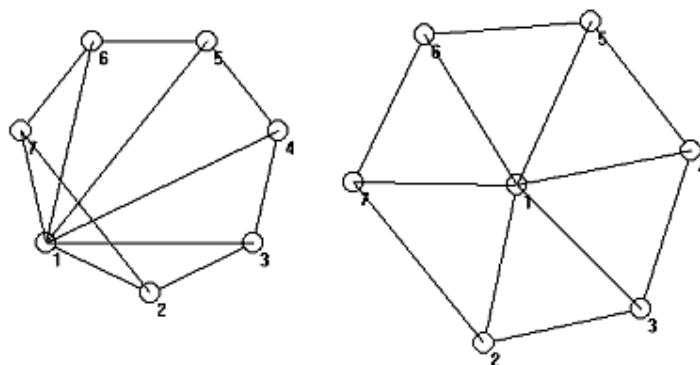


Рисунок 1 – Первый (слева) и последний шагу (справа) работы алгоритма

При тестировании рисование графа с 7 вершинами (рис. 1) заняло время – 153 (мкс), что является удовлетворительным результатом. Однако были обнаружены и слабые стороны данного алгоритма – он не всегда способен минимизировать количество пересечений ребер, также необходим подбор начальных параметров элементов системы для определения размеров, которые займет представление графа.

Метод прожига. Задача минимизации энергии не может быть решена аналитически в общем случае. Применим алгоритм, который моделирует процесс отжига системы. Температуру системы (степень подвижности вершин) будем характеризовать числом T . Со временем система «остывает», т.е. T уменьшается. В процессе одной итерации каждая вершина делает попытку переместиться на произвольное расстояние $r \leq T$ в случайном направлении. Если энергия нового состояния меньше энергии старого, то система переходит в это новое состояние.

Исходя из проблем, обнаруженных в ходе реализации пружинного алгоритма, в качестве основного критерия визуализации примем наименьшее количество пересечений дуг графа. Это и будет энергией системы. За температуру примем то расстояние, на которое может переместиться вершина. Начальное значение температуры выбирается исходя из следующего условия нахождения всех вершин в пределах видимости, поэтому самое большое расстояние, на которое вершина может переместиться, оставаясь видимой, будет равно наименьшей из величин Высота, Ширина (их выберем исходя из теоремы, которая гласит, что каждый плоский трехсвязный граф имеет выпуклый рисунок на сетке размерами $(N-2) * (N-2)$). Это необходимо для тех вершин, которые могут попасть за границу видимого прямоугольника. Для каждой вершины генерируем случайные координаты в пределах этого прямоугольника. Вычисляем значение энергии, количество пересечений дуг графа при такой раскладке графа. Далее, если энергия системы уменьшилась, мы запоминаем новое положение вершины и пробуем сгенерировать новые координаты для другой вершины. С полным проходом вершин температура системы уменьшается, т.е. расстояние, на которое можно двигать вершины, тоже уменьшается.

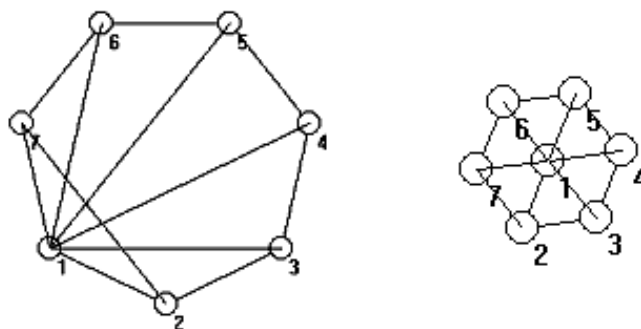


Рисунок 2 – Первый (слева) и последний (справа) шагу работы алгоритма

При тестировании рисование графа с 7 вершинами (рис. 2) заняло время – 219747 (мкс), что является плохим результатом. На основе полученных данных мы получаем хорошие результаты, но этот алгоритм вряд ли применим к большим графам (более 20-30 вершин) из-за своей трудоемкости.

Аналитические модели. Был разработан следующий аналитический алгоритм рисования графа:

1) Построение треугольного графа методом смещений (shifting) де Фрайсекса. Он основан на понятии канонического упорядочения вершин графа, располагает вершины планарного графа в целочисленных узлах системы координат, так что дуги представляются прямолинейными отрезками. Вершины раскладываются по порядку в соответствии со своими номерами, и при добавлении новой вершины уже разложенные могут смещаться так, чтобы избежать пересечения дуг

2) Производится ортогонализация графа путем ассоциации графа с потоковой сетью, где в качестве потоков представлены числа изгибов, либо размеры углов, образованных в вершинах граней. Вычисляется поток с минимальной стоимостью в сети и вычисляется ортогональное представление графа, ассоциированные с потоком с минимальной стоимостью.

3) Производится компактификация графа путем построения вертикальных и горизонтальных потоков ассоциированных для каждого из выделенных сегментов графа. Вычисляются потоки минимальной стоимости горизонтальный и вертикальный – и устанавливается длина каждого сегмента, равная потоку в соответствующей дуге.

Данный алгоритм будет реализован в ходе выполнения курсовой работы по предмету «Прикладные интеллектуальные системы и системы принятия решений».

Литература

1. Ioannis G. Tollis, Giuseppe Di Battista, Peter Eades, Roberto Tamassia. Graph Drawing: Algorithms for the Visualization of Graphs. – New Jersey, 1999.
2. T. Nishizeki, M. Rahman. Planar Graph Drawing. – Danvers, 2004.
3. M. Kaufmann, D. Wagner. Drawing Graphs Methods and Models. – Berlin, 2001.
4. M. Rahman T. Nishizeki. Orthogonal Drawings of Plane Graphs Without Bends, Journal of Graph Algorithms and Applications. – Vienna. – 2003. – P. 335.

УДК 004.6

ТЕСТИРОВАНИЕ ЗАДАЧ ОЛИМПИАДНОГО ХАРАКТЕРА

Рыжков А.А.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Силаев Н.В., доцент*

В данном сообщении обсуждается проблема создания программного комплекса автоматического тестирования решений задач олимпиадного характера.

Нам видится, что интерфейс системы тестирования может представляться в двух формах – в стиле веб-интерфейса, а также в стиле менее распространенном на практике – через комплекс программ «клиент-сервер». Мы предлагаем для рассмотрения тестирование через веб-интерфейс.

Тестирование через веб-интерфейс – это тестирование, которое может быть реализовано на языках php, ASP .Net, perl и др., сам же модуль сервера, обрабатывающий результаты, может представлять собой консольное приложение или WindowsForm приложение.

Рассмотрим ход проведения тестирования и требования, предъявляемые к программам тестировщикам:

- Основное, с чего начинается тестирование, – это удобное представление тестов, их редактирование и администрирование. На наш взгляд, среда комплекса тестирования должна обладать возможностями добавления нового теста.

- Среди принципиально важных возможностей администрирования, при компьютерном тестировании должны быть:

- возможность регистрации списков участников с заданием их логических имен и паролей входа;

- возможность установки времени начала, конца и «заморозки» тестирования;

- возможность работы с пользователями;

- возможность некоторых технических настроек и т.д.

- Среди принципиально важных возможностей пользовательского интерфейса при компьютерном тестировании должны быть:

- удобная авторизация;

- удобство и понятность интерфейса для получения нужной информации по ходу олимпиады.

- Отдельным моментом является защита тестировщика от взломов, т.к. может быть откомпилирован и запущен не безопасный для сервера код.

- Одной из актуальных проблем при компьютерном тестировании является нагрузка на сервер. Несколько пользователей могут одновременно послать задачу на тестирование. Для предотвращения ошибок, нужно организовать очередь проверки пользователей.

Система тестирования, написанная в рамках данного сообщения, базируется на взаимодействии веб-интерфейса написанного на php с сервером, написанным на C# WindowsForm. Для организации веб-сервера был использован сервер Denver.

Все взаимодействия между программой тестирования и пользовательским интерфейсом осуществляется через файл. Это не совсем удобно и небезопасно, поэтому далее большинство информации целесообразнее хранить в базе данных, используя к примеру MySQL. Обращение к MySQL реализовано как в php, так и в C#.

После запуска тестировщика, администратор олимпиады имеет возможность запустить веб-сервер (Denver), перезапустить и отключить его. Позднее будут введены возможности конфигурирования сервера напрямую из программы тестировщика, т.е. возможность изменения ip, порта, доменного имени.

Организация собственно олимпиадной работы для пользователя нам видится традиционной: получение условия – решение задачи - пересылка на сервер исходного текста решения (с указанием языка решения) – получение реакции от сервера по результату автоматического тестирования.

Тут возникает очень интересный вопрос – откуда берется компонента-модуль «тестировщик» и можно ли в дальнейшем подключать другие языки программирования? Если взять представляемый проект, то он имеет встроенный компилятор. В то же время тестировщик Contester, написанный российскими программистами, использует уже установленные в системе компиляторы, например платформа Dot Net имеет встроенные компиляторы. Для примера, в Framework для компиляции C# кода используется компилятор csc.exe. На настоящее время нам видится, что реализовать подключение новых компиляторов несложно, а главное, полезно и перспективно в силу того, что в практике про-

граммирования существует такое понятие, как «модный язык программирования», т.е. такой язык, который в настоящее время широко используется. Поскольку моды имеют тенденцию как приходить, так и уходить, то в целях создания наиболее технологичного комплекса, свободно перенастраиваемого на новые условия, нами планируется учесть и названное выше свойство возможности простого подключения внешних компиляторов.

Написанная среда тестирования представляет лишь начальный вариант проекта, который мы, при благоприятных условиях, собираемся развивать. Правила проведения олимпиад, некоторые нюансы будут изменяться и дополняться. В процессе эволюции нашего программного продукта мы надеемся, что он превратится из экзотического метода тестирования для отдельно взятых учебных заведений, в программный комплекс для массового использования с возможностью простой и быстрой его инсталляции.

УДК 53.0871.088

ИСПОЛЬЗОВАНИЕ ПАКЕТА “SIGNAL EXPRESS” ДЛЯ РЕГИСТРАЦИИ КИНЕТИКИ ГЕНЕРАЦИИ ЛАЗЕРА

Семёнов Г.А.

Алтайский государственный университет, г. Барнаул, Россия

Возможности программной среды LabVIEW на сегодня позволяют работать над решением огромного числа исследовательских задач, для чего создано множество различных пакетов и приложений, имеющих как специальный, так и универсальный характер. В рамках настоящей работы мы рассмотрим применение пакета “Signal Express” [1] для регистрации и обработки лазерных сигналов.

Пакет “Signal Express” содержит средства автоматического подключения внешних устройств ввода-вывода их конфигурирования, организации процедур измерения и анализа сигналов. Использование “Signal Express” позволяет быстро построить систему сбора и обработки информации. Рассмотрим прибор для измерения и отображения интенсивности лазерного излучения, его спектра мощности. Для преобразования лазерного сигнала используем 32-канальный АЦП NI 9205, преобразователь установлен в слотах системы сбора данных cDAQ-9172, которая соединена с компьютером посредством USB-порта.

Если на компьютере установлено программное обеспечение LabVIEW 8.2, после загрузки операционной системы будет выполнен автозапуск программы NI Device Monitor, которая предназначена для обнаружения подключённых устройств ввода-вывода (производства National Instruments).

После обнаружения устройства (в нашем примере это система сбора данных cDAQ-9172) программа предложит выбрать вариант используемого приложения. Выберем из предложенного меню “Начало измерений с использованием LabVIEW SignalExpress” (“Begin a Measurement with This Device Using NI LabVIEW SignalExpress”).

Программа визуализирует переднюю панель реальной системы сбора данных и все установленные в ней устройства ввода-вывода. В нашем примере это аналогово-цифровой преобразователь NI 9205. В случае наличия дополнительных устройств программа “Signal Express” предложит выбрать вариант измеряемой величины (например, напряжение, температура и т.д.).

В разделе конфигурирования выберем необходимое количество каналов, диапазон измеряемых интенсивностей (Signal Input Range), вариант подключения входов АЦП (Terminal Configuration) к источнику сигнала. В окне предварительного просмотра будет отображена осциллограмма сигнала, по которой можно судить о качестве преобразования сигнала.

Выбрав закладку “Калибровка” (“Calibration”), можно выполнить калибровку АЦП. В разделе “Временные установки” (“Timing Settings”) можно выбрать тип преобразования (непрерывное преобразование или преобразование определённого количества выборок сигнала), количество выборок для чтения и частоту преобразования.

Перейдём из установок в окно программы “Signal Express” и добавим процедуру вычисления спектра сигнала к уже имеющейся процедуре измерения интенсивности с помощью команды “Добавить шаг” (“Add Step”). В левой части экрана после закладки “Аналоговый ввод” (“Analog Input”) появится закладка “Спектр мощности” (“Power Spectrum”).

Для отображения сигнала и его спектра требуется два графических индикатора. Добавим второй индикатор командой “Добавить дисплей” (“Add Display”).

Для привязки индикатора к отображаемой функции спектра мощности нужно навести курсор на соответствующий индикатор и нажать правую кнопку мыши; затем выбрать последовательно “Сигналы” (“Signals”), “Добавить сигнал” (“Add Signal”), “Спектр” (“Spectrum”). На втором графическом индикаторе отобразится осциллограмма входного сигнала.

На рисунке 1 приведён пример графического представления спектра мощности хаотического сигнала (гауссовского белого шума), построенного в среде LabVIEW.

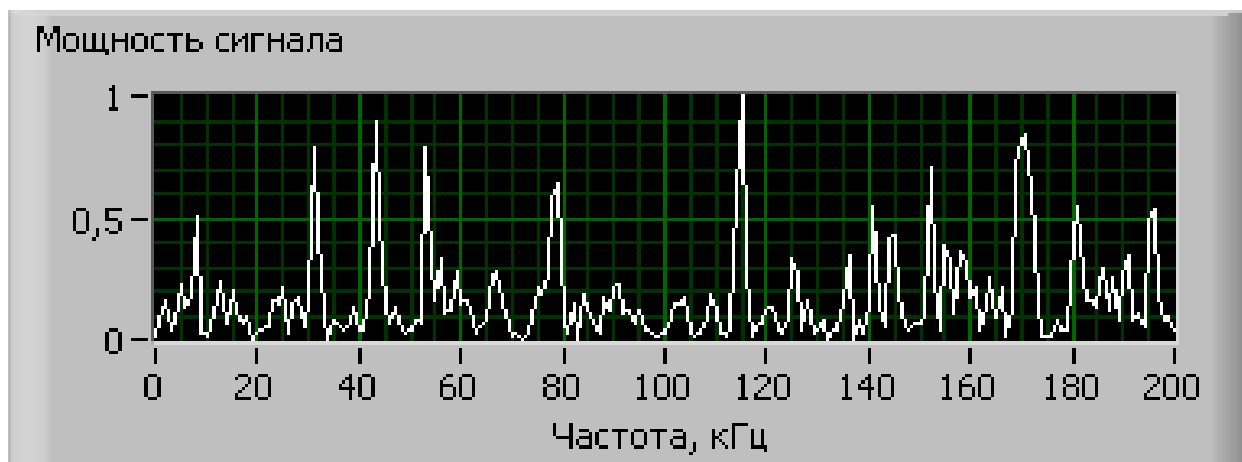


Рисунок 1 – Спектр мощности хаотического сигнала

Виртуальный прибор, созданный с помощью пакета “Signal Express” позволяет в режиме реального времени производить регистрацию и анализ лазерных сигналов в ходе физических экспериментов.

Литература

1. Виноградова, Н. А. Разработка прикладного программного обеспечения в среде LabVIEW: учебное пособие / Н. А. Виноградова, Я. И. Листратов, Е. В. Свиридов. – М.: Издательство МЭИ, 2005. – 48 с.

2. Останин, С.А. LabVIEW в биомедицине / С.А. Останин – Барнаул: Издательство ГОУ ВПО “Алтайский государственный университет”. 2009.–226с.

УДК 517

О МЕТОДАХ АНАЛИЗА НЕОДНОРОДНЫХ ВРЕМЕННЫХ РЯДОВ ГИДРОМЕТЕОРОЛОГИЧЕСКОЙ ИНФОРМАЦИИ

Сергиенко Е.Ю.

УО «Брестский государственный технический университет», г. Брест

Начиная с последнего десятилетия прошлого века, на территории Беларуси климатические аномалии стали наиболее сильно выражены и оказывали заметное влияние на экономическую ситуацию в стране [1]. Для диагностики климатических явлений важную роль играют данные наблюдений: надежность их получения и адекватность интерпретации результатов анализа.

Обычно глобальные климатические тренды, на основании которых делаются выводы об изменениях климата и о глобальном потеплении, выявляются при исследовании временного хода среднеглобальных значений геофизических параметров. Однако многие геофизические параметры характеризуются большой региональной изменчивостью (реанализы Cullather RJ, Bromwich D, Serreze M. 2000. *Climatologie*. V5. и Гидрометеоцентра) [2]. Пространственно-временная структура полей геофизических параметров характеризуется значительной неоднородностью, особенно во внетропических широтах, в которых и располагается вся территория Беларуси. Поэтому особую значимость приобретает изучение данных наблюдений в виде пространственно-временных полей. Пригодные для комплексного анализа данные в виде полей в достаточном объеме и с необходимым пространственно-временным разрешением охватывают лишь несколько последних десятилетий. Однако и за это время накоплено значительное количество информации, которая может и должна быть использована для анализа.

Описанные в работе методы будут применяться к данным об осадках и наводнениях на территории Беларуси за последние 50-60 лет. Целью работы является получение оценок и тенденций наступления неоднородностей во временных рядах гидрометеорологической информации и построение вероятностного прогноза.

Течение любых природных процессов нельзя предсказать однозначно, однако можно определить вероятности возникновения тех или иных явлений. Для этого необходимо выявить ряд статистических параметров, которые в комплексе достаточно полно характеризуют свойства исследуемой генеральной совокупности.

Теория цепей Маркова на данный момент глубоко разработана, что обуславливает актуальность ее применения при описании естественных процессов. Сравнительно небольшое число входящих в марковскую модель параметров обеспечивает относительную простоту ее применения по сравнению, например, с динамическими моделями. Для рассматриваемых процессов характерны случайные отклонения и взаимосвязь переменных во времени. Стохастические модели позволяют учитывать эти отклонения.

В дискретной модели наблюдения происходят в точечные моменты времени с заданной периодичностью. Марковская модель определяется вектором начальных и матрицей переходных вероятностей. Конечное множество значений объясняется тем, что в большинстве реальных процессов число возможных состояний системы ограничено.

Множество значений марковской цепи выбирается в соответствии с характером наблюдений и условиями задачи. Число условий задачи можно уменьшить, объединив некоторые из них или отбросив несущественные. Таким образом, многие марковские модели оперируют

двух- или трехфазным множеством состояний. При использовании подобных частных случаев, получаются более простые формулы, нежели для произвольного конечного числа состояний, а это дает возможность производить эффективные вычисления.

Актуальность марковского подхода при моделировании метеорологических процессов обусловлена определенными факторами. Во-первых, неоднородность рассматриваемой территории и временных рядов для каждого сезона и даже месяца не позволяет строить общую модель выпадения жидких осадков. Поэтому каждую достаточно однородную по рельефу и климатическим условиям область приходится исследовать отдельно. Во-вторых, практическая важность поставленных задач заключается в том, что на основе модельных реализаций и можно вычислять такие климатические характеристики, которые по имеющимся данным наблюдений оценить либо трудно, либо вообще невозможно.

В качестве модели берется простая (односвязная) однородная двоичная цепь Маркова, в которой состояние в данный момент времени зависит только от одного, непосредственно предшествующего состояния.

С помощью критерия, например, хи-квадрат, проверяется, насколько хорошо данная марковская модель описывает реальный процесс.

В рамках исследования проводится:

- анализ структуры марковских цепей, в частности, нахождение совместных распределений и первых моментов некоторых функционалов на траекториях цепи;
- рассмотрение марковской модели, описывающей исследуемые временные ряды, проверка ее адекватности и исследование серийной структуры с помощью полученных теоретических формул.

Методами исследования служат метод производящих функций, специальные методы теории марковских процессов, математического анализа, линейной алгебры и математической статистики.

Часто для анализа временных рядов также используется **преобразование Фурье**, которое дает разложение исследуемого временного процесса $f(t)$ в ряд по тригонометрическим функциям. Метод дает удобные для расчетов формулы и прозрачную интерпретацию результатов, однако имеет ряд некоторых недостатков. Преобразование, например, не отличает сигнал, являющийся суммой двух синусоид, от ситуации последовательного включения синусоид и др. Кроме того, исследуемые ряды также далеко не всегда удовлетворяют требованию периодичности и более того, как правило, заданы на ограниченном отрезке времени.

Основы **вейвлет-анализа** были разработаны в середине 80-х годов Гроссманом и Морле как альтернатива преобразованию Фурье для исследования пространственно-временных рядов с выраженной неоднородностью. В отличие от преобразования Фурье, локализирующего частоты, но не дающего временного разрешения процесса, и от аппарата δ -функций, локализирующего моменты времени, но не имеющего частотного разрешения, вейвлет-преобразование обладает самонастраивающимся подвижным частотно-временным окном. Таким образом, оно одинаково хорошо выявляет как низкочастотные, так и высокочастотные характеристики параметра на разных временных масштабах.

Подобно тому, как в основе аппарата преобразований Фурье лежит единственная функция $w(t)=\exp(it)$, порождающая ортонормированный базис пространства $L^2(0,2\pi)$ путем масштабного преобразования, так и вейвлет-преобразование строится на основе единственной базисной функции $y(t)$, принадлежащей пространству $L^2(\mathbb{R})$, т.е. всей числовой оси.

География приложений вейвлет-анализа очень широка. Например, в [3] на основе вейвлет-анализа выявлена существенная многомасштабность временных колебаний среднегодовой глобальной температуры воздуха за последние 150 лет. Делается вывод о некорректности использования традиционных средств тестирования стационарных случайных процессов без предварительного разделения колебаний на нестационарную (тренд) и осцилляторную части при изучении современных изменений климата. Прогнозируется возможная приостановка дальнейшего роста глобального потепления или, по крайней мере, его замедление.

Одним из направлений приложений вейвлет-анализа является исследование свойств фрактальных объектов различной природы и, в частности, определение фрактальной размерности, так, в [4] указанный подход используется при анализе временных рядов отклонений температуры от среднемесячных значений на метеорологической станции Фрунзе (Бишкек) за период 1931-1998 гг.

Литература

1. Логинов, В.Ф. Практика применения статистических методов при анализе и прогнозе природных процессов / В.Ф. Логинов, А.А. Волчек, П.В. Шведовский – Брест: Изд-во БГТУ, 2004.

2. Астафьева, Н.М. Региональная неоднородность климатических изменений / Н.М. Астафьева, М.Д. Раев, Н.Ю. Комарова.

3. Сонечкин, Д.М. Оценка тренда глобального потепления с помощью вейвлетного анализа / Д.М. Сонечкин, Н.М. Даценко, Н.Н. Иващенко – Известия РАН. Физика атмосферы и океана – № 2, – Т.33, 1997.

4. Козлов, П.В. Вейвлет-преобразование и анализ временных рядов / П.В. Козлов, Б.Б. Чен – Вестник КРСУ. – 2002. – № 2.

УДК 681.324

АНАЛИЗ ПОДХОДОВ К РЕШЕНИЮ И ТЕСТИРОВАНИЮ NP-ПОЛНЫХ ЗАДАЧ

Сечко Ю.Н., Матюшков Л.П.

УО «Брестский государственный технический университет», г. Брест

Развитие приближенных методов решения NP-полных задач идет в направлении их решения за приемлемое время с возможно меньшими отклонениями от оптимального ответа. Этот раздел исследований начинает становиться одним из важных в создании методов искусственного интеллекта (ИИ) для решения NP-полных задач, например, использование нейронной сети (НС) Хопфилда к решению задач комбинаторной оптимизации и др. [1] Сама природа задач фактически относит их к этому классу методов, поэтому большинство авторов вынуждено в целях ускорения решения и повышения его качества совершенствовать различные «эвристические» элементы и приемы, предлагать методы обучения НС особой конструкции, причем постепенно усиливается развитие тех из них, которые еще позволяют и динамическую настройку НС.

Классическим объектом из этой области является задача коммивояжера (ЗК), на которой можно проиллюстрировать все типичные подходы к решению NP-полных задач. В методах ее решения, как и других NP-полных задач, авторы стараются сохранить одно важное свойство: не терять возможности получить оптимальный ответ и ограничиться несколько худшим решением при лимите ресурсов (время ожидания результата (скорость обучения), переполнение памяти, ограничения в системах программирования и др.). Пространство поиска различных маршрутов, как указано в [1], равно $n!/(2n)$.

ЗК относится к классу задач комбинаторики, на которой можно проводить различные исследования, она также стала объектом приложения нейросетевых методов. В ее классической постановке требуется из исходного пункта объехать все остальные заданные пункты по кратчайшему маршруту, т.е. фактически за наименьшую стоимость, побывав в каждом пункте один раз, и вернуться в исходный. Расстояния (в общем случае стоимости проезда) между пунктами можно задать в виде квадратной матрицы A размерностью $n \times n$, где n – количество пунктов, a_{ij} – расстояние между пунктами i и j . В реальных ситуациях $a_{ij} \neq a_{ji}$ (например, одностороннее движение или пробки на дорогах). В общем случае ЗК не может опираться на геометрические понятия расстояния, поэтому основанные на них алгоритмы применять нельзя.

Основными точными методами решения ЗК являются метод полного перебора, общий метод ветвей и границ, метод Литтла, хорошо себя зарекомендовали и некоторые нейросетевые методы. В частности, Хопфилдом и Тенком была построена модель, которая для нескольких десятков городов в экспериментах позволила находить близкие к оптимальным решения за приемлемое время.

В оценке результатов решения разумно с позиций практики рассматривать различные конкурирующие методы как между собой, так и среди нейросетевых. Поскольку задача является NP-полной, можно найти оптимальное решение, перебрав все варианты, но такой подход не всегда осуществим из-за больших затрат времени даже при решении на мощных ЭВМ.

Общий метод ветвей и границ заключается в том, что можно не продолжать ветвление из данной вершины графа-дерева, если некоторая оценка в этой вершине достигнет порогового значения. Если принять за исходный пункт 1 начала движения корневую вершину графа-дерева, то любой вариант обхода всех пунктов будет определяться как сумма стоимостей конкретных ветвей графа по кольцевому маршруту. Тогда любая известная часть маршрута будет иметь точную оценку $F_{\text{факт}}$ (фактически пройденный путь до промежуточной вершины j) и оптимистическую оценку $F_{\text{опт}}$ для завершения маршрута. Ветвление из данной вершины можно не рассматривать, если ее суммарная оценка превысила пороговое значение или равна ему.

Оценка вершины j складывается из $F_{\text{факт}}$ и оценки $F_{\text{опт}}$, такой, что фактически оставшийся путь будет больше или равен ей. Для ускорения вычислений $F_{\text{опт}}$ можно считать равной сумме минимальных путей из тех вершин, которые еще не пройдены. Таким образом, в начальной вершине 1 $F_{\text{факт}} = 0$, а $F_{\text{опт}}$ равна сумме минимальных чисел в каждой строке входной матрицы A . Исходя из структур конкретных задач, можно принимать и другие функции, однако выбор во всех случаях зависит от человека. Пороговое значение $F_{\text{рек}}$ называется текущим рекордом, и оно равно длине лучшего маршрута, найденного на данный момент. Большое значение имеет выбор начального $F_{\text{рек}}$, т.к. чем меньше его значение, тем больше вершин отсекается при ветвлении. В качестве начального значения $F_{\text{рек}}$ можно принимать длину пути, найденного по принципу FIFO (первый пришел – первый обслуживаешься), т.е. стратегии всегда идти в ближайший пункт.

Используя эти соображения, можно описать следующий алгоритм:

- 1) Отыскать начальное значение $F_{\text{рек}}$
- 2) Из выбранной вершины i (начальная вершина $i=1$) построить все одношаговые пути в вершины j , которые не встречаются на пути из i в j , получить их оценки $F = F_{\text{факт}} + F_{\text{опт}}$, включить во множество V рассматриваемых вершин те из них, в которых $F < F_{\text{рек}}$.

3) Если множество V будет пустым, то $F_{рек}$ объявляется оптимальным ответом. Если среди вершин, построенных из i , нет таких, что их оценка $F < F_{рек}$, то выбрать из множества V вершину с минимальной оценкой и перейти к пункту 2. Если во множестве V присутствует несколько вершин с одинаковой оценкой, то выбирается вершина на наиболее дальнем ярусе от корня, а при наличии нескольких таких вершин на одном ярусе выбирается вершина в лексикографическом порядке.

4) Если построена концевая вершина графа и ее оценка $F < F_{рек}$, то F становится новым $F_{рек}$.

5) Выбрать из множества V при условии, что оно не пустое, вершину с минимальной оценкой и перейти к пункту 2.

Изложенный алгоритм легко запоминается и программируется, а также может быть использован при «ручном» решении. Однако он обладает следующими недостатками:

1) Выбранная проверка устраняет просмотр некоторых частей дерева, но на самом деле она может оказаться достаточно слабой и допускать глубокое проникновение внутрь дерева до того, как ветви отсекаются как бесперспективные [4].

2) Получаемое множество V вершин становится очень велико, прежде чем количество отсекаемых вершин превысит количество добавляемых в процессе решения. В таких случаях память переполняется (требуется 300 Мбайт и более).

3) Сложность выборки из большого ($\approx 10^6$ элементов) множества V вершины с минимальной оценкой. Поиск минимального элемента в массиве, его сортировка перед очередной выборкой, как выяснилось на практике, не эффективны. Наилучших результатов позволяет добиться использование сбалансированного бинарного дерева и сортировка подсчетом (при небольших числах во входной матрице A).

Одним из возможных способов преодоления такого рода трудностей является контроль за занимаемым промежуточными результатами объемом памяти и переходом на приближенный метод решения, если объем требуемой памяти близок к максимально допустимому значению. В этом случае можно взять 1-15 вершин с наименьшей оценкой и достроить из этих вершин маршруты по принципу FIFO, что приведет уменьшению занимаемого объема памяти, а также, возможно, позволит улучшить $F_{рек}$. Следует также отметить, что при таком подходе метод ветвей и границ может сходиться за заданное время, а при его превышении решение получается с некоторой погрешностью.

Другим способом является модификация оценочной функции с учетом реально пройденных путей. Можно попытаться корректировать $F_{опт}$ следующим образом: по достижении некоторого яруса дерева проанализировать 1-ый столбец матрицы A и найти минимальное значение пути в пункт 1 из тех вершин, которые еще не пройдены. Пусть этот путь будет равен $jmin$. Тогда вместо минимального числа в строке i , в которой на первой позиции находится $jmin$, можно будет прибавить к оптимистической оценке значение $jmin$. Пусть минимальное число в i -той строке равно $imin$. Тогда в худшем случае $imin=jmin$ и увеличение $F_{опт}$ не происходит. В остальных случаях $jmin$ будет больше $imin$, что приведет к росту $F_{опт}$, более обоснованному и быстрому отсечению вершин.

По сравнению с общим методом ветвей и границ метод Литтла показывает лучшие результаты и при этом остается точным, не приводит к переполнению памяти. Однако на некоторых примерах время решения задачи получается неприемлемым. Это связано с тем, что при наличии в матрице A большого количества одинаковых или очень мало отличающихся друг от друга элементов процедуры оценки нулей перестают быть эффективными, и много времени тратится на исследование маршрутов-дубликатов, уменьшение же временных затрат возможно при приближенном решении.

Таким образом, анализ известных методов показывает их уязвимость в зависимости от подбора исходных данных, что толкает исследователей к новому поиску методов решения с учетом особенностей прикладных задач и обеспечение гарантии получения ответа за приемлемое время. Нейросетевые подходы позволяют переложить сложности такого рода на ЭВМ за счет обучения соответствующих НС (Хопфилда и др).

Чтобы вынести практические рекомендации пользователю для его класса задач с имеющимися у них особенностями, разумно использовать процедуру тестирования для всех подходящих методов по общей схеме. Сложность проблемы составления тестов обнаруживается при проведении различных экспериментов. Ее суть состоит в том, что трудно найти эффективный способ конструирования тестовых примеров. Это может привести нас к мысли использовать методы ИИ и в таких целях. Наши эксперименты подтвердили трудность решения задачи создания исходных данных. В частности, универсальный метод генерирования матриц на основе реализации вероятностного закона оказался неэффективен. Это привело нас к выводу необходимости анализа среды заказчика и составления тестовых примеров с учетом ее особенностей.

Литература

1. Комарцова, Л.Г. Нейрокомпьютеры: учеб. пособие для вузов / Л.Г. Комарцова, А.В. Максимов. – М.: МГТУ им. Баумана, 2004. – 400 с.
2. Матюшков, Л.П. О решении методом ветвей и границ дискретных задач: сб. статей VII международной научно-практической конференции «Наука и образование в условиях социально-экономической трансформации общества» ч. 2 / Л.П. Матюшков, Г.Л. Матюшкова – г. Брест: Изд. Лавров С.Б., 2004. – С. 364-367
3. Матюшков, Л.П. Обучение использованию метода ветвей и границ в распределительной логистике: сб. статей «3-я международная научно-практическая конференция «Устойчивое развитие экономики: состояние, проблемы, перспективы» / Л.П. Матюшков, Г.Л. Матюшкова, Н.И. Зайцева – г. Пинск: Полесский ГУ, 2009 – С. 184
4. Рейнгольд, Э. Комбинаторные алгоритмы. Теория и практика. / Э. Рейнгольд. Ю. Нивергельт, Н. Део – М.: Мир, 1980 – 478 с.

УДК 621.865.8

АВТОНОМНЫЙ МОБИЛЬНЫЙ МИКРОКОНТРОЛЛЕРНЫЙ СТЕНД

Склипус, Д.Б.

УО «Брестский государственный технический университет», г. Брест

В современном мире происходит постоянное наращивание роботизации различных областей промышленности, сельского хозяйства, сферы обслуживания. В последнее время в рамках этой тенденции наблюдается также и увеличение доли использования автономных мобильных роботов, выполняющих задачи, связанные с перемещением в пространстве. Благодаря широкому распространению дешевых и простых в реализации средств связи и портативных вычислительных систем для данного направления робототехники прогнозируется скачкообразный рост. Ведущие профильные вузы реагируют на наличествующий и ожидаемый в будущем спрос введением дополнительных учебных курсов.

Лабораторные практикумы, касающиеся технологий управления мобильными роботами, чаще всего строятся на базе различных виртуальных сред моделирования [1, 2]. Изучение программирования электронных устройств невозможно без практических экспериментов, и это особенно важно в случае управления автономной мобильной системой. Последнее представляет из себя сложный аппаратный комплекс, в котором, помимо электронной составляющей, задействованы оптические, механические и, возможно, акустические устройства. Такой аппарат в своей физической реализации взаимодействует с широким спектром факторов и воздействий окружающей среды, подвержен ряду физических закономерностей, учет которых нехарактерен для систем моделирования и никогда не встречается в виртуальном окружении в полном объеме. Таким образом, разработка лабораторного мобильного стенда на базе программируемого микроконтроллера для решения учебных задач представляет несомненную практическую актуальность.

В данной работе представляется робот SRS 02, разработанный для нужд проведения лабораторных практикумов, связанных с программированием алгоритмов управления мобильным устройством. На рисунке 1 представлена фотография робота. Управление роботом может выполняться посредством встроенного микроконтроллера либо через персональный компьютер. Предусмотрены также два способа связи с компьютером: взаимодействие по интерфейсу RS232 через микросхему MAX232 для согласования уровней и телеуправление через модуль Bluetooth. В последнем случае микроконтроллер выполняет функции интерпретатора команд/данных.

Мобильность робота обеспечивают два электродвигателя, передающие крутящий момент на заднюю пару колес. В конструкции использованы коллекторные электродвигатели со встроенным редуктором, передаточное число которого имеет соотношение 50 к 1.

Напряжение питания составляет 7В. Источником питания служит набор из шести батарей типа АА. Для формирования напряжения питания интегральных схем применён преобразователь на базе микросхемы 7805.

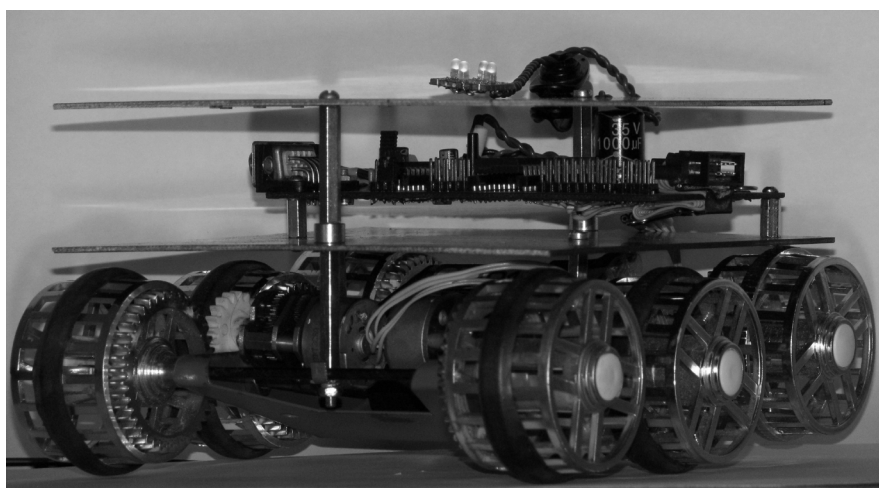


Рисунок 1 – Робот SRS 02

На рисунке 2 изображена основная плата робота - блок управления, который основан на микроконтроллере Atmega32. Микроконтроллер содержит 32 кбайт внутрисистемной программируемой флэш-памяти с функцией чтения во время записи, 1 кбайт ЭСППЗУ, 2 кбайт статического ОЗУ, 32 линии универсального ввода-вывода, 32 универсальных рабочих регистра, счетчик реального времени (RTC), два гибких таймера-счетчика с режимами сравнения и ШИМ, УСАПП, двухпроводной последовательный интерфейс,

ориентированный на передачу байт, 8-канальный 10-разрядный АЦП с опциональным дифференциальным входом с программируемым коэффициентом усиления, программируемый сторожевой таймер с внутренним генератором, последовательный порт SPI, испытательный интерфейс JTAG, совместимый со стандартом IEEE 1149.1, который также используется для доступа к встроенной системе отладки и для программирования. Загрузка кодов команд в память микроконтроллера осуществляется из персонального компьютера через внутрисхемный программатор JTAG. На плате предусмотрена возможность подключения разнообразных датчиков как разработанных для предыдущих моделей роботов [3, 4], так и новых, например, акустических.

Гибкость и многофункциональность разработанного мобильного стенда позволяет проводить множество практических экспериментов. Однако типовая задача для реализации управления мобильным устройством — это программа движения по заданному маршруту из точки А в точку Б. В предыдущих работах [4, 5] данная задача решалась нами за счёт нанесения чёрной линии на плоскость движения робота. Таким образом, робот мог двигаться по линии, придерживаясь маршрута, ограниченного ею и алгоритмом движения, заложенным в микроконтроллер. Однако такой подход ограничивает возможности автономного устройства, не позволяя двигаться в произвольном направлении, в т.ч. когда координаты точек А и Б являются переменными величинами.

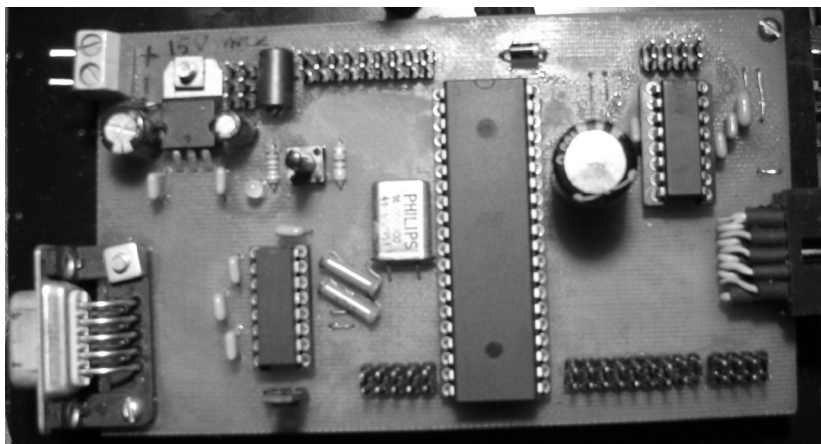


Рисунок 2 – Блок управления

Типичное решение проблемы позиционирования мобильного устройства с помощью системы GPS в данном случае неприменимо из-за слишком большой погрешности относительно размеров робота (~ 5 метров) и неустойчивой работы в помещении. Нами предложена гибридная система, использующая наблюдение за движением робота с помощью цифровой камеры и согласование данных с расчетными значениями перемещения робота. На основе показаний датчиков вращения колес робот может проехать около 30 см; после этого накопление погрешности достигает такого уровня, что расчет перемещения становится нецелесообразным. Поэтому, после обработки полученного с веб-камеры изображения, относительные координаты передаются роботу через Bluetooth-модуль. Рисунок 3 показывает расположение камеры и робота.

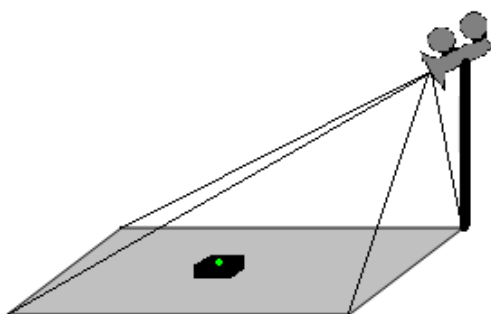


Рисунок 3 – Схематическое расположение робота и камеры наблюдения

Для облегчения позиционирования робота на снимке на него крепится маркер - яркий светодиод зелёного цвета.

После обработки изображения и получения координат маркера x и y в пикселях, требуется провести их пересчёт в реальные декартовы координаты на плоскости движения робота, выполняемый по следующей формуле:

$$X = Z \cdot \frac{y \cdot p_{32} \cdot p_{13} - y \cdot p_{33} \cdot p_{12} + p_{22} \cdot p_{33} \cdot x - p_{22} \cdot p_{13} - p_{23} \cdot x \cdot p_{32} + p_{23} \cdot p_{12}}{-x \cdot p_{31} \cdot p_{22} - x \cdot p_{32} \cdot p_{21} - p_{11} \cdot p_{32} \cdot y + p_{11} \cdot p_{22} + p_{12} \cdot p_{31} \cdot y - p_{12} \cdot p_{21}}$$

$$Y = -Z \cdot \frac{-x \cdot p_{31} \cdot p_{23} + x \cdot p_{33} \cdot p_{21} - p_{11} \cdot y \cdot p_{33} + p_{11} \cdot p_{23} + p_{13} \cdot y \cdot p_{31} - p_{13} \cdot p_{21}}{-x \cdot p_{31} \cdot p_{22} - x \cdot p_{32} \cdot p_{21} - p_{11} \cdot p_{32} \cdot y + p_{11} \cdot p_{22} + p_{12} \cdot p_{31} \cdot y - p_{12} \cdot p_{21}}$$

где X и Y – искомые координаты, а значения векторов p получаются в процессе калибровки камеры.

Программное обеспечение для обработки сигналов с камеры разработано средствами языка программирования C# в среде Microsoft Visual Studio 2008. Экранная форма представлена на рисунке 4.

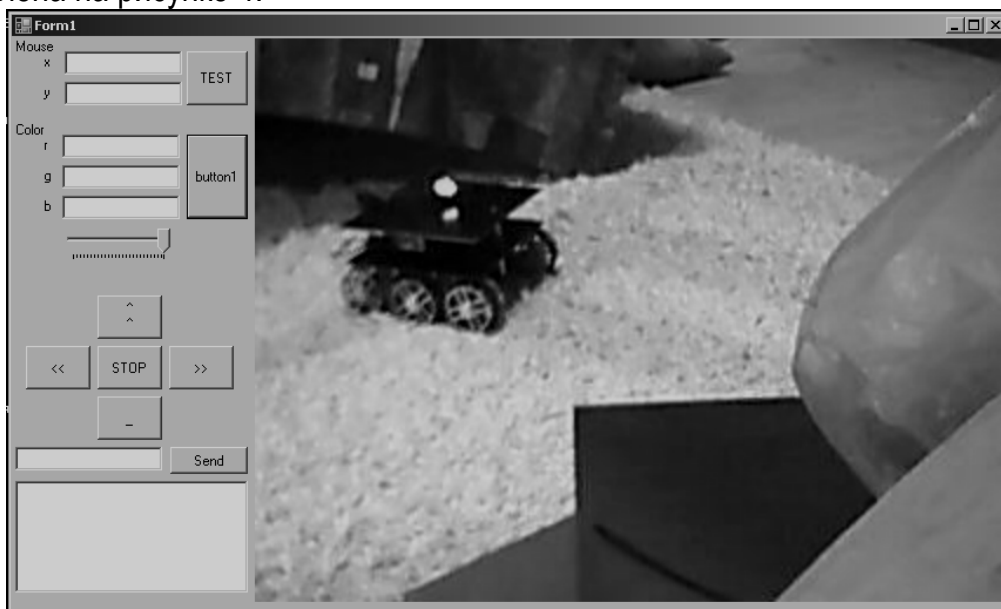


Рисунок 4 — Программное обеспечение работы с веб-камерой

Литература

1. Robotics Toolbox for MATLAB. 10/10/2005. <http://www.ict.csiro.au/downloads/robotics/>
2. Собственный робот в Robotics Studio. Создание 3D-модели. 27/07/ 2009. <http://insiderobot.blogspot.com/2009/06/robotics-studio-3d.html>
3. Склипус, Д.Б. Микроконтроллерная лаборатория / Д.Б. Склипус // Информационные технологии управления в экономике 2006: материалы респ. науч.-практ. конф. Брест, 2006. – С. 35.
4. Склипус, Д.Б. Микроконтроллерная лаборатория / Д.Б. Склипус // сб-к тезисов по секции «Информатика» Междунар. конф. «VI Колмогоровские чтения». Москва, 2006. – С. 26.
5. Склипус, Д.Б. Алгоритм движения по черной линии для автономного мобильного робота / Д.Б. Склипус // сб-к конкурсных научных работ студентов и магистрантов. – Брест, 2009. Ч. 1. – С. 122–125.

ДРЕВОВИДНЫЙ АЛГОРИТМ РАЗБИЕНИЯ СЛОВ РУССКОГО ЯЗЫКА НА СЛОГИ НА ОСНОВЕ СТАТИСТИЧЕСКИХ ДАННЫХ

Согоян А.Л.

УО «Брестский государственный технический университет», г. Брест

Огромная часть информации, воспринимаемая человеком, имеет словесный вид. К нему относятся тексты, находящиеся во всемирной паутине, информация, публикуемая в газетах, журналах и других средствах передачи информации. Для того, чтобы тексты выглядели на страницах Интернет-ресурса или газеты в привлекательном и удобном виде, как одно из средств форматирования используют расстановку переносов. В основе расстановки переносов лежит разбиение слов на слоги. В связи с тем, что людям приходится работать с огромными объемами текста, возникает необходимость автоматизировать процесс разбиения слов на слоги – разработать алгоритм, который будет качественно и быстро производить разбиение слова на слоги и, при этом, не будет требовать много памяти.

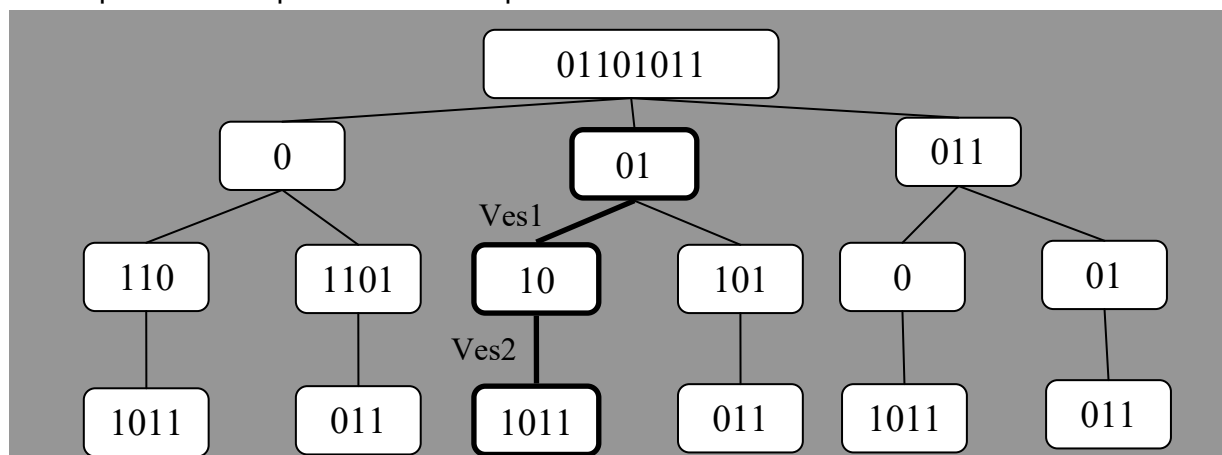
Для сбора весовых коэффициентов цепочек слогов используем имеющийся словарь уникальных русских слов, состоящий из 95725 слов. С помощью существующего онлайн-сервиса quittance.ru производим разбиение всего словаря русских слов на слоги. Данный онлайн-сервис использует в разбиении алгоритм Ляна-Кнута для автоматической расстановки переносов [1].

Полученный словарь русских слов, слоги которых разделены дефисами, преобразуем по правилу замены букв: гласные заменяем нулем, согласные единицей. Формируем массив вхождений соседних слогов в слова словаря.

В результате статистического анализа словаря русских слов были получены, для цепочки из 2 слогов – 289 вариантов связей, для цепочки из 3 слогов – 1092. Для цепочки из двух слогов наибольшую частоту имеет «10-10», а для цепочки из трех слогов «10-10-10».

На вход функции подается слово из русского языка. Буквы в слове заменяются цифрами 1 или 0. Для согласной буквы – 1, для гласной – 0. В результате имеется бинарное слово, которое представляется в виде дерева слогов. На рисунке изображено дерево для разбиения на слоги слова «алгоритм». В бинарном представлении данное слово будет выглядеть как «01101011». Количество нулей в бинарном слове соответствует количеству уровней в дереве, не считая самого верхнего.

Получив дерево всех возможных разбиений слова, начинаем его обход по каждой ветви. Имея статистическую матрицу связей слогов, считаем вес всего разбиения. Производим суммирование весов связей слогов на соседних уровнях. Разбиение с максимальным весом имеет наибольшую вероятность [2]. Исходя из этого, за правильное разбиение принимается разбиение бинарного слова на слоги с наибольшим весом.



Количество \ Длина цепочки	2	3
Слов	95 725	95 725
Правильных разбиений	53 623	64 607
Неправильных разбиений	42 102	31 118
Эффективность	56,02%	67,49%

Используя исходный словарь русских слов, на основе которого формировался массив коэффициентов связей цепочек слогов, произведем разбиение слов словаря и сравним с эталонным разбиением. В таблице представлены данные по количеству слов, участвующих в разбиении на слоги, количеству правильных результатов, а также эффективности алгоритма, использующего цепочки слогов длиной 2 и 3.

Среди слов, которые не были правильно разбиты программой на основе описываемого алгоритма, преобладают имеющие два или более корня в своем составе. С увеличением длины цепочки эффективность древовидного алгоритма разбиения слов русского языка растет. В связи с тем, что алгоритм не дает точного результата для всех слов, его целесообразно использовать в комбинации с другими методами. Так, например, использовать одновременно разбиения слов при вычислении веса разбиения с цепочками длиной два, три и более. А как результат – выбрать некоторую оптимальную комбинацию.

Литература

1. Donald E. Knuth. Digital typography. CSLI Lecture Notes, no. 78. Stanford, 1999.
2. Яглом, А.М. Вероятность и информация / А. М. Яглом, И. М. Яглом – М.: Наука, 1973.

УДК 004.514.62

ИСПОЛЬЗОВАНИЕ ПЕРИФЕРИЧЕСКОГО ЗРЕНИЯ ПРИ НАВИГАЦИИ МЕЖДУ СТРАНИЦАМИ В ИНТЕРНЕТ-БРАУЗЕРЕ

Тавониус К.А.

УО «Брестский государственный технический университет», г. Брест

В [1] приведено сравнение с лабиринтом ориентирования в современном программном интерфейсе, когда пользователь не имеет возможности видеть одновременно, хотя бы схематично, изображение всего рабочего пространства. В основном такой подход вызван ограниченностью аппаратных ресурсов персонального компьютера, не позволяющих задействовать большие площади для вывода информации.

В последнее время все большее внимание уделяется попыткам использовать уменьшенный масштаб изображений, не находящихся в фокусе работы пользователя, для увеличения наглядности и интуитивности интерфейса [2]. Не в последнюю очередь оживление в данной области связано с ростом разрешающей способности дисплеев, делающей более информативной технологию применения уменьшенных изображений объектов для предварительного просмотра (previews или thumbnails).

В последнее время масштабные преобразования начинают применяться в средствах Интернет-навигации. Однако их использование на сегодняшний день ограничено использованием сильно уменьшенных изображений веб-страниц в качестве ярлыков. В данной работе представлено дальнейшее развитие модели использования масштабирования веб-страниц, основанное на использовании аналогии периферического зрения.

Периферическое зрение, называемое часто также боковым или палочковым зрением, играет важную роль в ориентировании человека в окружающей среде [3]. Периферическое зрение определяется полем зрения. Полем зрения неподвижного глаза называют пространство, которое одновременно видит глаз (или оба глаза), фиксируя определенную точку неподвижным взором при неподвижном положении головы.

Границы поля зрения зависят от индивидуальных особенностей наблюдателя. Этим можно объяснить то, что в литературе приводятся различные значения размеров поля зрения. Нормальные границы поля зрения одного глаза таковы [5]: по горизонтали: к виску – $90 - 100^\circ$, к носу – $50 - 60^\circ$ (всего $140 - 160^\circ$); по вертикали: вверх – $50 - 60^\circ$, вниз $60 - 75^\circ$ (всего $110 - 135^\circ$).

Поле, одновременно охватываемое двумя глазами, по горизонтали несколько больше 180° и по вертикали около 120° . При вращении глаз наибольшее отклонение зрительных осей составляет $\pm 45 - 50^\circ$.

На рисунке 1 показаны горизонтальные углы поля зрения, сектора ПЗ отмечены 2 и 3. Согласно [6], зона ПЗ более чем вдвое перекрывает угол обзора прямого зрения.

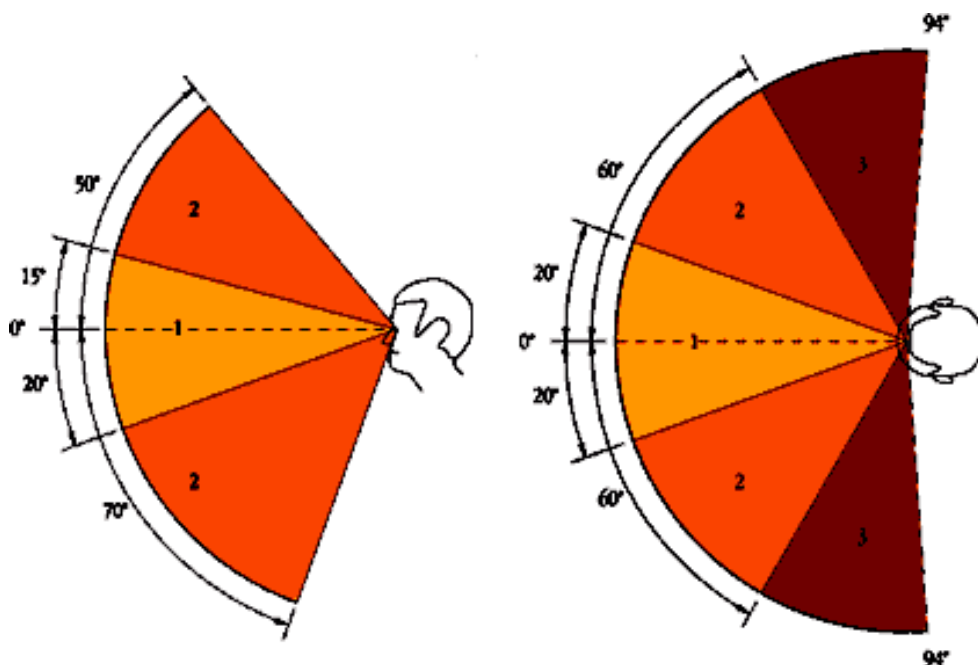


Рисунок 1 – Углы поля зрения

Несложно заметить, что поле зрения в горизонтальной плоскости больше, чем в вертикальной. Этим фактом давно и успешно пользуются производители мониторов. По этой причине в предлагаемой модели мы рассматриваем именно периферическое зрение в горизонтальной плоскости.

Концепция периферического зрения смоделирована нами в виде деления рабочей области на три зоны: центральную 2 и периферийные 1 и 3, как указано на рисунке 2. В зоне 2 отображается текущая Интернет- страница. В зоне 1 находятся миниатюры страниц, с которыми мы работаем в текущий момент. В зоне 3 могут быть расположены эскизы недавно закрытых страниц. Размер эскизов в зонах 1 и 3 при удалении к краю экрана, уменьшается согласно формуле (1).

$$\Delta x = \begin{cases} C, x \in 2 \\ C \cdot \alpha(x) & x \in 1, x \in 3 \end{cases} \quad (1)$$

где Δx – размер зерна (расстояние между центрами отдельных точек изображения), C – константа, определяющая единичный размер зерна, соответствующий области 2, $\alpha(x)$ – функция пространственных искажений [4], монотонно возрастающая и имеющая область значений, принадлежащую интервалу (0; 1).

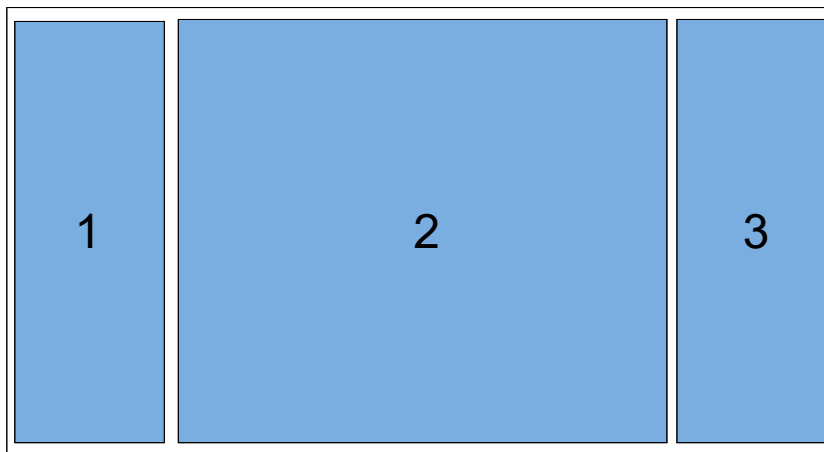


Рисунок 2 - Зонирование окна

Площадь миниатюры равняется $M \times N$ (ширина и высота области вывода миниатюр); таким образом, зона размером $W \times H$ может вместить максимум $\frac{W}{M} \times \frac{H}{N}$ пиктограмм.

Если A (число миниатюр для размещения) превышает данное, то $A - \frac{W}{M} \times \frac{H}{N}$ миниатюр попадает в алгоритм периферического сжатия. Тогда в периферической зоне они размещаются в масштабе $1:2^n$, где n – номер столбца. Нумерация столбцов начинается с нуля, и счёт идёт от зоны 2 (в зоне 1 – слева направо, а в зоне 3 – справа налево). Таким образом, в первом столбце зоны 1 – размер миниатюр равен $M \times N$, а в n -ом столбце зоны 3 – $\frac{M}{2^n} \times \frac{N}{2^n}$. На практике миниатюры с размерами менее 64×64 являются неинформативными. Поэтому для большей читаемости представляется целесообразным ограничить уменьшение размера миниатюр, сведя правило пересчета для n -ого столбца к $\max\left(\frac{M}{2^n} \times \frac{N}{2^n}, \frac{M}{2^2} \times \frac{N}{2^2}\right)$, а миниатюры с размерами $\frac{M}{2^2} \times \frac{N}{2^2}$ будут отображать не эскизы страниц, а их логотипы. При этом в n -й столбец попадает $\min\left(2^n \cdot \frac{W}{M}, 2^4 \cdot \frac{W}{M}\right)$ миниатюр.

На этапе анализа было выдвинуто три варианта реализации модели, изображённых на рисунке 3. Линейная модель 3-а проста в реализации и вполне привычна для пользователя. Модель с увеличением масштаба миниатюр слева направо и сверху вниз (т.н. модель со скоплением), представленная на 3-б, исключает эффект «висячих» столбцов и обеспечивает более эффективное использование площади окна. Также предусмотрено дальнейшее развитие модели 3-б, с двумя вертикальными зонами увеличения масштаба (рис. 3-в), что теоретически должно более полно задействовать поле зрения, однако может усложнить поиск необходимой миниатюры.

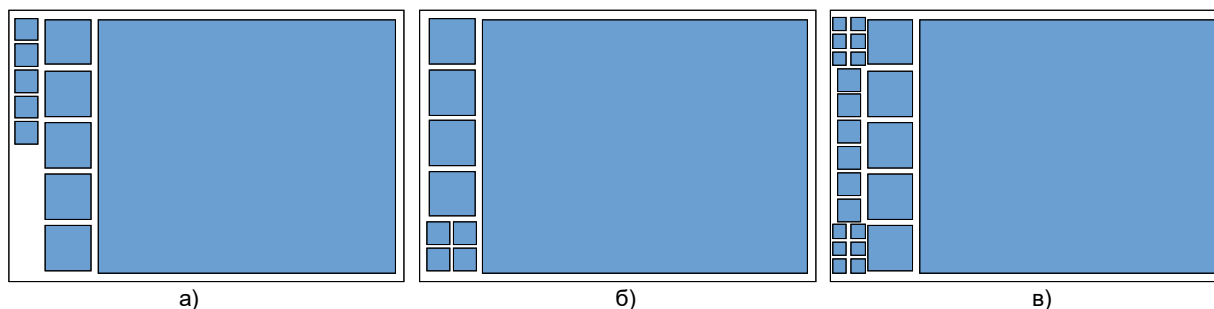


Рисунок 3 – Модели концепции периферического зрения, а) – линейная модель, б) – модель со скоплением, в) – усовершенствованная модель с двумя скоплениями

Литература

1. Раскин, Дж. Интерфейс: новые направления в проектировании компьютерных систем / Дж. Раскин – СПб.: Символ-Плюс, 2003. – 272 стр.
2. Борушко, И.Н. Применение модели периферического зрения в графическом интерфейсе пользователя. Современные информационные компьютерные технологии: сб. науч. ст. / И.Н. Борушко, Е.В. Гоманова, Д.А. Костюк. Гродно: ГрГУ, 2006. – С. 22–27.
3. Хацевич, Т.Н. Медицинские оптические приборы: Физиологическая оптика: учебное пособие / Т.Н. Хацевич – Новосибирск: СГГА, 1998. – 98 с.
4. Прэтт, У. Цифровая обработка изображений / У. Прэтт – М.: Мир, 1982. – Т. 1. – 310 с.
5. Yerry, M.A., Shephard, M.S. A Modified Quadtree Approach to Finite Element Mesh Generation // IEEE Comput. Graph., and Appl., «Construction of Polyhedra and Geometric Modeling», CAD Group Document № 100, Cambridge University Computer Laboratory.
6. Парфенов, В. Дизайн аудио-видео среды. Оптические расчеты. 2004.
<http://www.homeav.ru/HomeCin/OptiCalc.html>

УДК 336.762

ДИНАМИЧЕСКОЕ ОПРЕДЕЛЕНИЕ ОБЪЕМА СОВЕРШАЕМОЙ СДЕЛКИ КАК СПОСОБ ОПТИМИЗАЦИИ МЕХАНИЧЕСКОЙ ТОРГОВОЙ СИСТЕМЫ

Тараканова О.М.

УО «Гродненский государственный университет имени Янки Купалы», г. Гродно

Моделирование финансовых рынков является новой быстроразвивающейся областью прикладной математики. Это связано с повсеместным развитием финансовых рынков – фондового и валютного.

Механическая торговая система является универсальным и достаточным инструментом для принятия решений на рынке валют, который является одним из наиболее эффективных и надежных, позволяет при ограниченном риске достигать высоких значений уровня доходности на капитал. Основным принципом работы механической торговой системы является осуществление торговых операций, исходя из четких правил (разработанного ранее алгоритма), основанных на показаниях индикаторов технического анализа.

Для прогнозирования движения рынка (изменений цены валюты, объема сделок и открытого интереса) применяется технический анализ, основанный на информации, полученной за предыдущее время. В основном он сводится к исследованию графиков прежнего движения валютных цен, чтобы предсказать их возможное дальнейшее движение. Технические аналитики занимаются поиском в поведении цен сигналов, которые могли бы указать им на изменения в рыночной психологии и направлении тенденции. К таким сигналам относятся разворотные свечные комбинации.

Механическая торговая система разработана на основе метода «японские свечи». Алгоритм принятия решений по определенным разворотным свечным комбинациям описан в работе [1].

Первым этапом оптимизации механической торговой системы является применение формул теории вероятности: формулы полной вероятности и Байеса, для определения прибыльности или убыточности торгового советника. По формуле полной вероятности вычисляем вероятность получения прибыли при определенном наборе свечных комбинаций, а по формуле Байеса вычисляется вероятность получения прибыли при открытии позиции для каждой из рассмотренных свечных моделей.

Торговая система была протестирована на различных валютных парах на временных интервалах М1, М5, М15, Н1 и Н4. Результаты вычислений дают возможность оптимизировать разработанную механическую торговую систему, устранив из рассмотрения убыточные комбинации.

Следующий этап оптимизации торговой системы основан на динамическом определении объема совершаемой сделки. Это значит, что объем лота зависит от объема имеющихся средств на балансе текущего счета. Принято решение рисковать половиной имеющихся средств, выставляя соответствующий размер лота программным путем, в то время, как ранее эта величина была константой, равной единице, задаваемой пользователем.

Формула для расчета объема лота имеет вид:

$$L = \frac{F \cdot P}{100 \cdot Pl \cdot S}, \quad (3)$$

где L – новый лот, F – свободные средства на счете, P – процент денег от свободных средств, которые могут участвовать в сделке, в случае разрабатываемой программы – это 30%, Pl – стоимость одного лота, S – шаг изменения размера лота, в нашем случае шаг равен 0,1.

Динамическое определение объема совершаемой сделки положительно сказывается на изменении состояния счета, т.е. если количество денежных средств на балансе увеличивается, то и вероятность получения большей прибыли возрастает, т.к. мы задействуем половину имеющихся свободных средств. Вместе с тем, если торговая система проигрывает – баланс уменьшается, то и объем проигранных денежных средств сокращается, убыток минимизируется за счет динамического уменьшения размера лота.

Оптимизированная механическая торговая система была протестирована на валютных парах EUR/USD (евро/доллар США), EUR/GBP (евро/британский фунт), USD/CAD (доллар США/канадский доллар) на временных интервалах М1, М5, М15, Н1 и Н4.

Рассмотрим пример для данных по валютной паре GBP/USD с обновлениями каждые 15 минут (М15) на промежутке с 15.09.2008 по 16.10.2008, прибыль составила 16225\$ при начальном депозите 10000\$.

Литература

1. Тараканова, О.М. «Японские свечи» как метод технического анализа для разработки торговых стратегий / О.М. Тараканова, Н.В. Марковская // Анализ, моделирование, управление, развитие экономических систем: труды II Международной Школы-симпозиум – Симферополь, 2008. – С. 274-276.

2. Тараканова, О. М. Моделирование механических торговых стратегий для работы на валютном рынке / О.М. Тараканова, Н.В. Марковская // Современные информационные компьютерные технологии: сб. науч. ст. В 2 ч. / ГрГУ им. Я. Купалы; редкол.: Е. А. Ровба, А. М. Кадан. – Гродно: ГрГУ, 2008. – Ч. 2 – С. 144-147.

ОБЩАЯ АРХИТЕКТУРА КЛАССОВ ДОСТУПА К БАЗАМ ДАННЫХ

Тейт А.А.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Силаев Н.В., доцент*

В настоящее время абсолютное большинство программ, разрабатываемых программистами-прикладниками, ориентировано на работу с базами данных. Вместе с тем, программирование баз данных сопряжено с определенными особенностями, которые могут быть незнакомы разработчику, работавшему прежде лишь в области программирования математических задач или системного программирования.

Дело в том, что в математических задачах источником данных служит консоль и, что чрезвычайно редко, файл. Аналогичным образом при разработке системных приложений информация чаще всего извлекается либо из файлов, либо из системного реестра. Все эти источники данных объединяет одна особенность – за исключением тех случаев, когда файл расположен в сети, доступ к ним сравнительно быстрый и все команды доступа жёстко стандартизированы, т.к. являются функциями-членами стандартных классов используемого языка программирования.

В отличие от описанных выше случаев, программирование баз данных имеет свои особенности, не всегда очевидные для человека, который только начинает с ними работать.

Во-первых, скорость доступа к информации, содержащейся в базе данных, всегда существенно ниже, чем к файловой информации.

Во-вторых, язык запросов SQL, на котором программируются операции доступа к базе данных, существенно отличается по своей структуре от обычного программного кода. Вне зависимости от языка, на котором ведётся разработка программы (это может быть C++, Java, PHP, C#), перед программистом встаёт задача чёткой и аккуратной реализации классов для работы со сложной базой данных, в которой содержится множество таблиц и связей между ними.

Нередко начинающий программист пишет код доступа к базе данных «напрямую», прямо в обработчике нажатия кнопки, загрузки страницы и т.п. Заметим, что такой способ сложен, не технологичен и соответствует современным требованиям стиля программирования. Дело в том, что при подобном подходе, код, вызывающий базу данных, оказывается «рассеянным» по всей прикладной программе, так как в этом случае информация из базы данных вызывается напрямую в те моменты, когда она бывает затребована алгоритмом.

Недостатки такого подхода в следующем: в случае, если в базу данных систематически вносятся изменения, а без подобных операций практические задачи немислимы, программисту необходимо модифицировать команды SQL исходного кода программы. В связи с этим возникает очередная проблема: если даже исходный код программы размещён в едином файле, легко пропустить какой-либо из обращений к базе данных, даже если приложение сравнительно небольшое. Помимо этого, традиционно достаточно сложной является проблема тестирования работы всех операций вызова. Дело в том, что для большинства языков программирования, включая и языки высокого уровня, команда SQL является дополнительной переменной типа String. Поэтому практически все компиляторы (исключая некоторые расширения для C#, сделанные в Microsoft Visual Studio .Net 2008) не осуществляют даже синтаксической проверки SQL кода.

В связи со сказанным, для языков высокого уровня, включающих средства объектно-ориентированного программирования, разумно выделять классы, работающие с базой данных, в отдельный модуль. Для языков без поддержки наследования классов (таких, к примеру, как PHP), где доступ к базам данных реализован «напрямую» (то есть, HTML-таблица отображается сразу), самым удачным вариантом будет отдельный класс, в котором записаны все вызовы операций, разбитые по таблицам, для которых они применяются. Нами введен, в частности класс, содержащий, например, такие методы-функции, как добавление элемента, обновление данных элемента, получение Item по ID, получение всех Item, преобразование строки БД в элемент типа Item, преобразование строки БД в коллекцию элементов типа Item.

Что же касается таких языков, как C#, в которых для отображения содержимого базы данных уже имеются готовые компоненты, распознающие не только таблицы, но и связанные списки, то для них очень удобно использовать целый набор из классов (назовем его DBManager) для доступа к базе данных. Методы подобного класса получают непосредственный доступ к базе данных и возвращают необходимые таблицы, как элементы типа DataTable. Прочие же классы (назовем их %Type%DBAccessor, где Type – название типа, который необходимо возвращать), получают команды непосредственно из классов, отвечающих за обработку и отображение данных. Они возвращают связанные списки того типа, к которому принадлежат запрашиваемые данные.

Описанный подход нами реализован при построении системы тестирования общего назначения. Подобная система требует обработки большого объема разнородной информации (списки типов тестирования (текущее, экзаменационное, тематическое), списки разделов тестирования, списки тем тестирования, списки вопросов теста и соответствующих ответов-подсказок, списки групп тестируемых, списки результатов тестирования для каждого участника этого процесса), хранящейся в базе тестирования. Упомянутая система тестирования, развивающая традиции прежних систем, разработанных и использующихся на математическом факультете БрГУ им. А.С. Пушкина, в настоящее время проходит комплексные испытания и внедрение в учебный процесс. По материалам наших исследований были сделаны доклады на ряде научных конференций от университетских до республиканских.

УДК 681.01(075.6)

ALLFUSION PROCESS MODELER 7 (BPWIN) КАК СРЕДСТВО ФУНКЦИОНАЛЬНОГО МОДЕЛИРОВАНИЯ БИЗНЕС-ПРОЦЕССОВ

Тимова Т.И.

УО «Белорусский государственный экономический университет», г. Минск

AllFusion Process Modeler 7 (BPwin) – мощный инструмент моделирования, который используется для анализа, документирования и реорганизации сложных бизнес-процессов. Модель, созданная средствами BPwin, позволяет четко документировать различные аспекты деятельности - действия, которые необходимо предпринять, способы их осуществления, требующиеся для этого ресурсы и др. Таким образом, формируется целостная картина деятельности предприятия - от моделей организации работы в маленьких отделах до сложных иерархических структур. При разработке или закупке программного

обеспечения модели бизнес-процессов служат прекрасным средством документирования потребностей, помогая обеспечить высокую эффективность инвестиций в сферу ИТ. В руках же системных аналитиков и разработчиков BPwin - еще и мощное средство моделирования процессов при создании корпоративных информационных систем (КИС).

Модели BPwin дают основу для осмысления бизнес-процессов и оценки влияния тех или иных событий, а также описывают взаимодействие процессов и потоков информации в организации. Неэффективная, высокочатратная или избыточная деятельность может быть легко выявлена и, следовательно, усовершенствована, изменена или устранена в соответствии с общими целями организации.

Внешние обстоятельства зачастую вынуждают вносить изменения в деятельность организации. Последствия этих изменений должны быть тщательно изучены и осмыслены перед тем, как система будет переделана с их учетом. BPwin может помочь пользователю на протяжении всего цикла, предоставив возможность оптимизировать бизнес-процесс, которого коснутся эти изменения.

С помощью BPwin пользователь может сделать свою работу более продуктивной. Действия и другие объекты создаются буквально несколькими щелчками мыши, а затем легко отбуксированы в нужное место. Интерфейс BPwin, выполненный в стиле "проводника", облегчает навигацию и редактирование сложных процессов с иерархической структурой. Развитые возможности изменения масштаба представления позволяют быстро найти и сосредоточиться на необходимой для работы части модели процесса.

BPwin позволяет:

- Обеспечить эффективность операций, рассматривая текущие бизнес-операции через мощные инструменты моделирования.
- Совершенствовать бизнес-процессы, формулируя и определяя альтернативные реакции на воздействия рынка.
- Быстро исключать непродуктивные операции, легко и интуитивно сопоставляя операционные изменения. Неэффективные, неэкономичные или избыточные операции могут быть легко выявлены и, следовательно, улучшены, изменены или вовсе исключены - в соответствии с целями компании.

Функциональные возможности BPwin:

- Поддержка нескольких нотаций. Многочисленные технологии моделирования и поддержка разнообразных нотаций обеспечивают всесторонний и детальный анализ и проектирование в сложной операционной среде.
- Интуитивно-понятный графический интерфейс, который быстро и легко осваивается, позволяет сосредоточиться на анализе самой предметной области, не отвлекаясь на изучение инструментальных средств.
- Анализ показателей затрат и производительности. BPwin полностью поддерживает методы расчета себестоимости по объему хозяйственной деятельности (функционально-стоимостной анализ, ABC).
- Свойства, определяемые пользователем (UDP). BPwin позволяет настроить сбор дополнительной существенной для вашего бизнеса информации с помощью UDP - свойств, определенных пользователем.
- Организационные графики (диаграммы Swim Lane, диаграммы FEO, диаграммы дерева узлов). BPwin поддерживает точное описание ролей, которые определяют и распределяют по категориям задачи или работы внутри бизнес-процессов.
- Методы контроля корректности модели. Наличие контекстно-зависимой панели инструментов, невозможность создания в модели некорректных связей и т.д.

- Интерфейс к средствам имитационного моделирования. Имитационное моделирование позволяет исследовать результаты изменений в динамике.

- Документальный центр проекта. Модель бизнес-процессов в VPwin может быть организована как документальный центр проекта, когда все связанные с проектом документы, инструкции, программные приложения могут быть открыты непосредственно из среды модели.

- Работа с моделями бизнес-процессов из собственных программных приложений.

- Интеграция процессов/данных.

- Собственный генератор шаблонов отчетов.

VPwin имеет достаточно простой и интуитивно понятный интерфейс пользователя. При запуске VPwin по умолчанию появляется основная панель инструментов, палитра инструментов (вид которой зависит от выбранной нотации) и, в левой части, навигатор модели — Model Explorer.

При создании новой модели возникает диалог, в котором следует указать, будет ли создана модель заново или она будет открыта из файла либо из репозитория ModelMart, затем внести имя модели и выбрать методологию, в которой будет построена модель.

VPwin поддерживает три методологии — IDEF0, IDEF3 и DFD, каждая из которых решает свои специфические задачи. В VPwin возможно построение смешанных моделей, т. е. модель, может содержать одновременно диаграммы как IDEF0, так и IDEF3 и DFD. Состав палитры инструментов изменяется автоматически, когда происходит переключение с одной нотации на другую.

Литература

1. Маклаков, С. VPwin, ERwin. CASE- средства разработки информационных систем.
2. Вендров, А. CASE-технологии. Современные методы и средства проектирования информационных систем.
3. Калянов, Г. Теория и практика реорганизации бизнес-процессов.

УДК 372.016

ОБЩИЕ ТРЕБОВАНИЯ ПОДГОТОВКИ ЭЛЕКТРОННЫХ ПРЕЗЕНТАЦИЙ

Ткач С.Н.

УО «Брестский государственный университет имени А.С.Пушкина», г.Брест

Мультимедиа-технологии — это одно из перспективных направлений информатизации современного общества. На сегодняшний день обязательной частью многих мероприятий, в том числе и процесса обучения, является мультимедийная презентация.

Мультимедиа – это взаимодействие визуальных и аудиоэффектов под управлением интерактивного программного обеспечения с использованием современных технических и программных средств, они объединяют текст, звук, графику, фото, видео в одном цифровом представлении.

Основной принцип действия мультимедийных презентаций – это влияние на визуальное мышление слушателей. Действительно, результаты исследований показывают, что эффективность слухового восприятия информации составляет 15%, зрительного — 25%, а их одновременное включение в процесс обучения повышает эффективность восприятия до 65%. Интерес к данным технологиям возрастает еще и в связи с быстро становящимися доступнее новыми техническими средствами.

Термин «презентация» одновременно обозначает процесс представления материала аудитории и сам документ, в котором содержатся текст и мультимедиа-материалы. От выступающего с презентацией требуется не только умение создать документ и грамотно оформить его, но и искусство донести его содержание до аудитории.

Среди положительных результатов применения электронных презентаций, как для слушателей, так и для организатора презентации, можно выделить следующие [1]:

- повышается информативность и эффективность материала при его изложении, ввиду того, что у слушателей задействованы зрительный и слуховой каналы восприятия;
- увеличивается выразительность, наглядность и зрелищность излагаемого материала;
- снижается интенсивность труда организатора презентации во время представления материала, поскольку часть функций заменяется готовыми электронными презентациями.
- исключается вероятность ошибочной трактовки мыслей.

Создание презентаций позволяет упорядочить мысли, классифицировать материал, вскрыть «узкие» места.

Воздействие материала на слушателей во многом зависит от степени иллюстративности устного материала. Визуальная насыщенность материала делает его ярким, убедительным, способствует лучшему усвоению и запоминанию. Ввиду того, что презентация представляет весь отобранный и подготовленный материал в концентрированном, сжатом виде, то отбираемый материал должен включать различные виды наглядности [2]:

Наглядность I рода – это уникальные информационные материалы, которые слушатели видят непосредственно в результате показа (картины, рукописи, видеофрагменты, звукозаписи, фото, изображения окон при изучении визуального программирования, листинги программ, результаты работы алгоритмов и проч.).

Наглядность II рода – это графическое отображение (схемы, графики, чертежи и т.п.), вытекающее из необходимости систематизации и структурного представления материала.

Наглядность III рода – это мультимедийная наглядность, которая позволяет не только сочетать наглядности I и II рода, но и значительно расширить и обогатить их возможностями введением фрагментов мультимедиа с целью повышения эффективности работы с моделями изучаемых объектов, явлений или процессов для исследования их свойств, визуализации изучаемых явлений, процессов и взаимосвязей между объектами.

Среди недостатков использования электронных презентаций следует особо отметить высокую трудоемкость подготовки данных материалов, так как процесс их создания – это всегда большая, кропотливая и сложная работа.

Одним из основных компонентов дизайна презентации, на наш взгляд, является учет физиологических особенностей восприятия цветов и форм. К наиболее значимым из них относят:

- текстовый материал должен быть написан достаточно крупным шрифтом (не менее 24 размера). На одном слайде не следует размещать много текстовой информации (не более 2 определений или не более 5 тезисных положений). При этом, лучше воспринимается текст, выровненный «по центру». При выборе шрифта следует учитывать, что прописные буквы тяжелее, чем строчные: отношение толщины основных штрихов шрифта к их высоте составляет 1:5;

– сочетание двух цветов – цвета знака и цвета фона – существенно влияет на зрителя: некоторые пары цветов не только утомляют зрение, но и могут привести к стрессу. Составление цветовой схемы презентации должно начинаться с выбора двух главных функциональных цветов, которые используются для фона и обычного текста. Цвета на всех слайдах одной презентации должны быть одинаковы, что создаст у обучающегося ощущение связанности, стильности, комфортности [3];

– фон является элементом заднего (второго) плана. Он должен выделять, оттенять, подчеркивать информацию, находящуюся на слайде, но не заслонять ее. Белое пространство признается одним из сильнейших средств выразительности, однако малым признаком стиля. Любой фоновый рисунок повышает утомляемость глаз обучаемого и снижает эффективность усвоения материала. Более воспринимаемыми сочетаниями цветов шрифта и фона являются следующие: белый на темно-синем, белый на пурпурном, черный на белом, желтый на синем;

– большое влияние на подсознание человека оказывает мультипликация. Ее воздействие гораздо сильнее обычного видео. Четкие, яркие, быстро сменяющиеся картинки легко влияют на подсознание. Причем, чем короче воздействие, тем оно сильнее. Любой движущийся объект понижает восприятие обучающихся, оказывает сильное отвлекающее воздействие, нарушает динамику внимания;

– включение в качестве фонового сопровождения звуков (песен, мелодий) способствует утомляемости слушателей, рассеиванию внимания и снижению производительности обучения.

Следует выделить наиболее общие требования к средствам, формам и способам представления содержания учебного материала в электронной презентации [4]:

- сжатость и краткость изложения, максимальная информативность текста;
- объединение семантически связанных информационных элементов в целостно воспринимающиеся группы;
- каждому положению (каждой идее) должен быть отведен отдельный абзац текста;
- основная идея абзаца должна находиться в самом начале (в первой строке абзаца). Это связано с тем, что лучше всего запоминаются первая и последняя мысли абзаца;
- предпочтительнее использование табличного (матричного) формата представления материала, который позволяет представить материал в компактной форме и наглядно показать связи между различными понятиями;
- при проектировании характера и последовательности представления учебного материала должен соблюдаться принцип стадийности: информация может разделяться в пространстве (одновременное отображение в разных зонах одного слайда) или во времени (размещение информации на последовательно демонстрируемых слайдах);
- абсолютно недопустимы орфографические, грамматические или стилистические ошибки;
- графика должна органично дополнять текст. Динамика взаимоотношений визуальных и вербальных элементов и их количество определяются функциональной направленностью излагаемого материала. Известно, что глаз и мозг способны работать в двух режимах: в режиме быстрого панорамного обзора с помощью периферийного зрения и в режиме медленного восприятия детальной информации с помощью центрального зрения. При работе в режиме периферийного зрения система глаз-мозг почти мгновенно воспринимает большое количество информации, при работе в режиме центрального зрения - производится тщательный последовательный анализ. Следовательно, когда человек читает текст, да еще с экрана компьютера, мозг работает в замедленном режи-

ме. Если же информация представлена в графическом виде, то глаз переключается во второй режим, и мозг работает быстрее. Именно поэтому в учебных презентациях желательно свести текстовую информацию к минимуму, заменив ее схемами, диаграммами, рисунками, фотографиями, анимациями, фрагментами фильмов.

Учет указанных особенностей оформления презентации в значительной степени влияет на эффективность представленной в ней информации.

Таким образом, на сегодняшний день внедрение электронных презентаций в нашу жизнь на различных ее этапах является одним из методов повышения качества представления и восприятия информации. Несмотря на слабую техническую оснащенность учебных заведений повышение трудоемкости учителей и преподавателей, данные информационные технологии имеют ряд неоспоримых преимуществ.

Использование электронных презентаций позволяет представить материал как систему ярких опорных образов, наполненных исчерпывающей структурированной информацией в алгоритмическом порядке. В этом случае задействуются различные каналы восприятия слушателей, что позволяет заложить информацию в память не только в фактографическом, но и в ассоциативном виде. Подача материала в виде мультимедийной презентации сокращает время обучения. Их использование целесообразно на любом этапе изучения материала и на любом этапе обучения.

Литература

1. Можяева, Г.В. Как подготовить мультимедиа курс?: Методическое пособие для преподавателей / Г.В. Можяева, И.В. Тубалова; под ред. В.П. Демкина. – Томск: Изд-во Том. ун-та, 2008. – 75с.

2. Емельянова, В.И. Использование информационных технологий в преподавании спецдисциплин: Фестиваль педагогических идей «Открытый урок» [Электронный ресурс] // Официальный сайт газеты «1 сентября» – Режим доступа <http://festival.1september.ru/>. – Дата доступа: 1.10.2009.

3. Петроченко, Е.В. Влияние цвета на восприятие информации человеком: Фестиваль педагогических идей «Открытый урок» [Электронный ресурс] // Официальный сайт газеты «1 сентября» – Режим доступа <http://festival.1september.ru/>. – Дата доступа: 5.10.2009.

4. Ефремов, В.С. Методические рекомендации по созданию электронных презентаций / В.С. Ефремов. – Томск. 2005 – 10 с.

УДК 517

МЕТОД АНАЛИЗА ИЕРАРХИЙ В УПРАВЛЕНЧЕСКИХ ЗАДАЧАХ

Ткачёнок Е.М.

УО «Белорусский государственный экономический университет», г.Минск

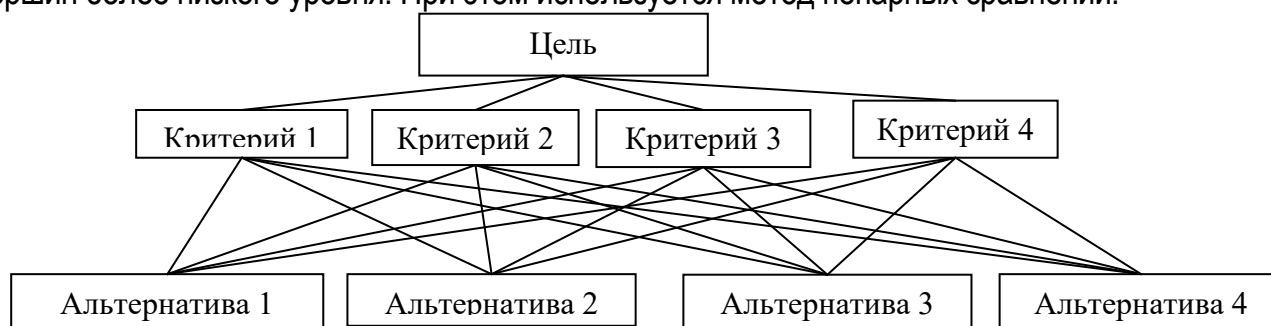
Очень часто управленческие проблемы имеют несколько вариантов решений. Зачастую, выбирая одно решение из множества возможных, лицо, принимающее решение, руководствуется только интуитивными представлениями. Вследствие этого принятие решения имеет неопределенный характер, что сказывается на качестве принимаемых решений. В последнее время для принятия решений в задачах, имеющих неопределённость, используется метод анализа иерархий, опирающийся на многокритериальное описание проблемы, был предложен Томасом Саати.

Суть данного метода заключается в количественном выражении качественных суждений. Проблема структурируется в виде иерархии, вершиной которой является цель, а последующими уровнями – соответствующим образом определённые критерии и альтернативы. Исследование сводится к последовательности попарных сравнений критериев и попарных сравнений альтернатив.

Применение метода анализа иерархий очень эффективно при решении недетерминированных задач, задач, в которых параметры имеют лишь качественную оценку, например: задача оценки недвижимости, задача выбора руководителя, задача выбора муниципального транспорта и т.д.

Для определения наиболее оптимальной альтернативы выделяются параметры оценки проекта, при помощи которых выбирается наилучший вариант. Таким образом, в МАИ основная цель исследования и все факторы, в той или иной степени, влияющие на достижение цели, распределяются по уровням в зависимости от степени и характера влияния. На первом уровне иерархии всегда находится одна вершина – цель проводимого исследования. Вторым уровнем иерархии составляют факторы, непосредственно влияющие на достижение цели. При этом каждый фактор представляется в строящейся иерархии вершиной, соединенной с вершиной 1-го уровня. Третий уровень составляют факторы, от которых зависят вершины 2-го уровня. И так далее. Этот процесс построения иерархии продолжается до тех, пока в иерархию не включены все основные факторы или хотя бы для одного из факторов последнего уровня невозможно непосредственно получить необходимую информацию.

По окончании построения иерархии для каждой материнской вершины проводится оценка весовых коэффициентов, определяющих степень ее зависимости от влияющих на нее вершин более низкого уровня. При этом используется метод попарных сравнений.



Интенсивность влияния параметра оценивается по шкале относительной важности: 1 - одинаковая значимость, 3 - некоторое преобладание значимости одного фактора перед другим (слабая значимость), 5 - существенная или сильная значимость, 7 – очень сильная или очевидная значимость, 9 – абсолютная значимость, 2,4,6,8 - промежуточные значения между соседними значениями шкалы.

Выбор шкалы определялся следующими требованиями:

- Шкала должна давать возможность улавливать разницу в чувствах людей, когда они проводят сравнения, различать как можно больше оттенков чувств, которые имеют люди.

- Эксперт должен быть уверенным во всех градациях своих суждений одновременно.

Для оценки альтернатив по выбранным критериям применяется метод попарных сравнений, при помощи которого определяются веса критериев и оценки альтернатив по каждому критерию, а также произведение оценок по строке (Π_j) и корень n -ой степени

из произведения экспертных оценок по строке ($\sqrt[n]{I_j}$). После чего определяется оценка альтернативы как результат отношения корня n-ой степени из суммы оценок по строке к их сумме по всем оценкам.

Для определения значимости оценок проектов и параметров производится расчет сопоставимости результатов: находится оценка сопоставимости результатов (λ), которая должна быть больше 4, и фактический индекс сопоставимости (ИС_ф), который впоследствии сравнивается с табличным значением (ИС_т).

При незначимых оценках предложенные альтернативы следует переоценить и произвести перерасчет значимости снова.

Затем определяется общая оценка - взвешенная сумма альтернатив с учётом оценки каждой альтернативы по каждому критерию и веса каждого критерия. В результате высшая оценка соответствует наиболее оптимальному проекту.

Экономическая задача представляет собой определение наилучшей альтернативы из 4 предложенных, основываясь на 4 параметрах оценки. Примером такого задания может служить выбор наиболее оптимального инвестиционного проекта, где в качестве предложенных альтернатив выступают различные варианты вложения средств.

Проекты представлены 4 альтернативными странами вложения инвестиций: Германия, Греция, Болгария, Россия.

Условиями оценки проектов представлены следующие параметры: срок окупаемости инвестиций, инвестиционный климат в стране-реципиенте, норма доходности инвестиций и социальный эффект.

Параметр	Срок окупаемости, лет	Инвестиционный климат	Норма доходности, %	Социальный эффект
Проект				
Германия	3	Отличный	12	Сильный
Греция	5	Удовлетворительный	16	Умеренный
Болгария	2	Неудовлетворительный	15	Относительный
Россия	4	Хороший	20	Отсутствует

Определить, в какую страну наиболее выгодно вкладывать средства.

Для решения данной задачи применима написанная мною на языке C++ программа по методу анализа иерархий, которая в качестве базовых допущений предполагает ограниченное число проектов (4) и параметров (4).

Входные данные представляют собой экспертные оценки по шкале относительной важности, а выходные - агрегированные оценки проектов по всем критериям, с помощью которых выбирается оптимальная альтернатива (большей оценке соответствует наилучший вариант).

При запуске программы перед пользователем появляются редактируемые поля, в которые он вводит наименования оцениваемых проектов и параметров оценки. Затем активируется кнопка «Далее» и программа переходит к оценке значимости параметров. Пользователь вводит оценки параметров в ячейки таблицы на пересечении оцениваемого параметра с тем, с которым он сравнивается. Программа автоматически проставляет единичные оценки в ячейки, где параметр сравнивается сам с собой (Рис.1).

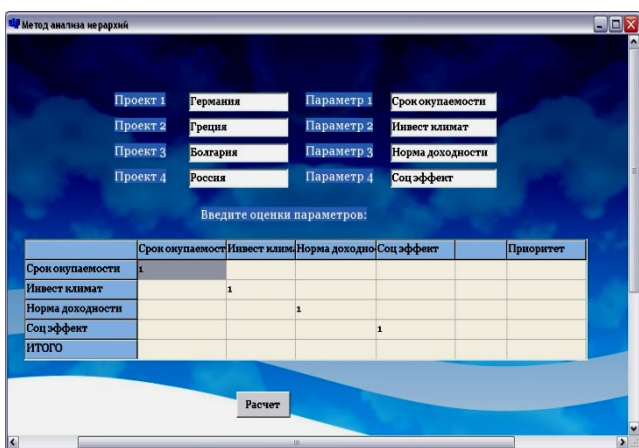


Рисунок 1 – Оценка значимости параметров

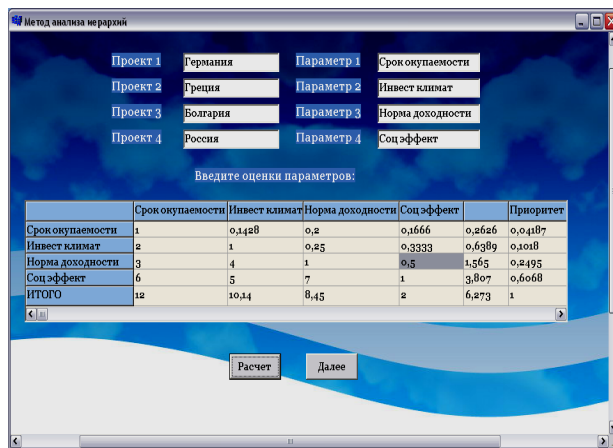


Рисунок 2 – Расчет приоритетов параметров

При нажатии кнопки «Расчет» происходит расчет промежуточных значений. А затем, путём нахождения удельного веса каждого значения в их сумме, определяются приоритеты параметров, которые в сумме должны давать единицу, что проверяется на пересечении ячеек «ИТОГО» и «Приоритет» (Рис.2).

При переходе по кнопке «Далее» программа начинает поочередную оценку предложенных проектов по каждому из параметров, начиная с первого. Пользователь вводит оценки одних проектов относительно других по определённому параметру и по кнопке «Расчет» получает значения приоритетов проектов по данному параметру.

Программа автоматически проверяет значимость оценок проектов, путем расчета коэффициента сопоставимости результатов λ . Где λ равна сумме произведений суммы оценок одного проекта относительно других на соответствующие значения приоритетов. Условием значимости оценок является $\lambda \geq 4$. Если данное условие выполняется, то программа выводит значение оценки сопоставимости результатов, в противном случае – рекомендует переоценить проекты по данному параметру.

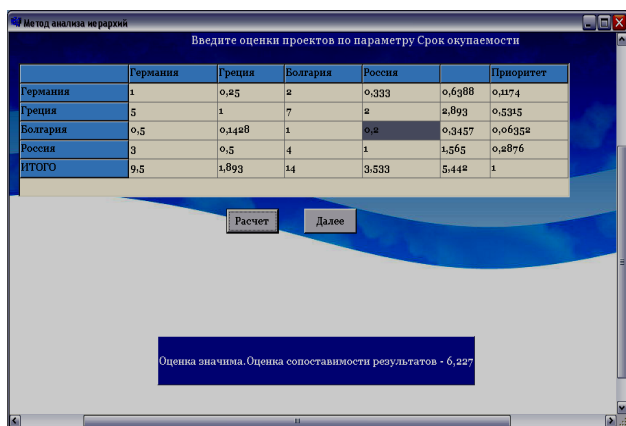


Рисунок 3 – Оценка проектов по параметру

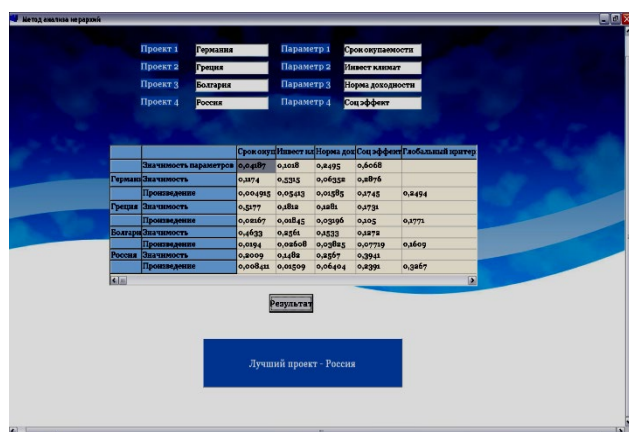


Рисунок 4 – Вывод результата

При нажатии кнопки «Далее» данная операция повторяется 4 раза (соответственно количеству параметров), после чего рассчитывается результат, т.е. находится глобальный критерий оценки проектов и определяется его максимальное значение и соответствующий ему проект.

Таким образом, метод анализа иерархий применим для решения управленческих задач различного вида, в том числе характеризующихся значительной степенью неопределенности.

Написанная программа «Метод анализа иерархий» позволяет быстро и точно определить наилучшую альтернативу действий при помощи оценок четырех имеющихся вариантов и четырех параметров значимости данных проектов.

Простой и удобный пользовательский интерфейс позволяет без труда понять принципы работы программы и приступить к её применению. Кроме того, при её использовании значительно сокращаются затраты времени на проведение сложных расчетов вручную и, что более важно, исключается возможность ошибки, допущенной человеком, решающим задачу.

УДК 004.514.62

ПРИМЕНЕНИЕ ДИАГОНАЛЬНОЙ ПРОКРУТКИ В ГРАФИЧЕСКОМ ИНТЕРФЕЙСЕ МОБИЛЬНЫХ УСТРОЙСТВ

Тухто Н.Н.

УО «Брестский государственный технический университет», г. Брест

В настоящее время наблюдается оживление в поиске инновационных элементов графического интерфейса, не в последнюю очередь связанное с распространением мобильных устройств с сенсорным экраном, рассчитанных на управление пальцами и накладывающих новые требования на размер и расположение элементов.

Несмотря на свое название, сенсорные экраны не задействуют чувство осязания пользователя в полной мере, создавая вместо этого иллюзию восприятия. Пользователь не ощущает обратной реакции при нажатии на кнопку экрана. Безусловно, это до некоторой степени освобождает от физических усилий, но взамен мы в значительной степени отказываемся по от одного из каналов восприятия информации. Пользуясь сенсорным экраном, нельзя набрать телефонный номер с закрытыми глазами. Интерфейс не дает тактильных ощущений.

Сенсорные экраны, кроме того, накладывают ограничения на формфакторы, что существенно усложняет использование интерфейса. На мобильных устройствах необходима площадка диаметром как минимум 9 мм, чтобы, коснувшись ее пальцем, пользователь мог выполнить какую-либо операцию. Поэтому выпускается множество устройств с крупными интерфейсными элементами и сокращается число малогабаритных устройств. С другой стороны, при работе с планшетными ПК использование тактильного интерфейса сопряжено с частыми изменениями положения рук, а это неудобно. Необходимость коснуться экрана, чтобы вызвать меню, во многих случаях замедляет работу.

Ряд исследований, касающихся повышения эффективности графических интерфейсов с сенсорным управлением, посвящен поиску более эффективных элементов управления прокруткой. Кроме изменившихся аппаратных ресурсов, данный интерес связан с тем, что изначально традиционным средствам организации прокрутки присущ ряд недостатков. Как малоинформативный элемент интерфейса, полоса прокрутки должна занимать минимум рабочего пространства и располагаться на его периферии. Однако это делает доступ к ней крайне невыгодным по закону Фитса [1, 2] и весьма затрудненным в случае мобильного устройства с управлением пальцами.

Рассмотрим подробнее закон Фитса и его неэффективность в данной ситуации. Предположим, что пользователь перемещает курсор к кнопке, изображенной на экране. Длина прямой линии, соединяющей начальную позицию курсора и ближайшую точку целе-

вого объекта, определяется в законе Фитса как дистанция. На основе данных о размерах объекта и дистанции закон Фитса позволяет найти среднее время, за которое пользователь сможет переместить курсор к кнопке.

В одномерном случае, при котором размер объекта вдоль линии перемещения курсора обозначается как S , а дистанция от начальной позиции курсора до объекта – как D (рис. 1), закон Фитса формулируется следующим образом: $\text{Время (мс)} = a + b \cdot \log_2\left(\frac{D}{S} + 1\right)$.

Константы a и b устанавливаются опытным путем по параметрам производительности человека.



Рисунок 1 – Расстояния, которые используются в законе Фитса для определения времени, необходимого для перемещения курсора к цели

Вычисляемое время отсчитывается от момента, когда курсор начинает движение по прямой линии, до момента, когда пользователь щелкает мышью по целевому объекту.

Признаком хорошего интерфейса является рациональное использование всей его площади. Ее пространство можно и нужно использовать эффективнее. Так, визуальный дизайнер Эдвард Тафт разработал принципы отображения информации, среди которых первыми тремя являются следующие:

- данные следует показывать прежде всего остального;
- следует максимально увеличивать долю чернил, используемых для отображения данных;
- следует максимально уменьшать долю чернил, которые не используются для отображения данных.

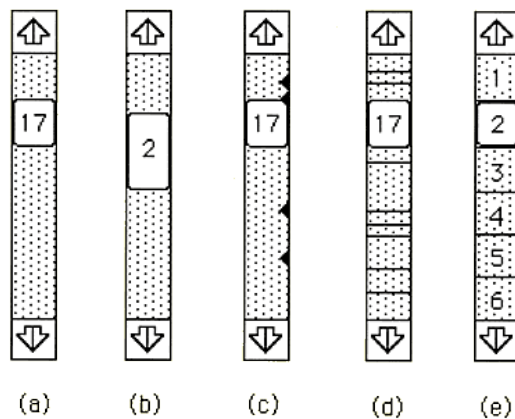
Однако для более быстрого перемещения курсора по закону Фитса необходимо увеличить размер S целевого элемента управления, что зачастую приводит к неэффективному расходованию площади экрана. Особенно заметен ущерб полезной площади вывода информации в случае полосы прокрутки, т.к. параметром S в подавляющем большинстве случаев является ее ширина. Возникает противоречие: узкая полоса прокрутки труднодоступна (при управлении мышью – из-за закона Фитса, при тактильном управлении – из-за площади контактного пятна), а широкая – крайне неэффективно расходует площадь экрана.

Тем не менее, полоса прокрутки – неотъемлемая часть большинства современных приложений. Хотя этот интерфейсный элемент управления используется уже очень давно, его функциональность и эргономическая ценность по-прежнему могут быть улучшены за счет некоторых добавлений.

Полоса прокрутки обладает как прямой функциональностью (пользователь перемещается по документу с ее помощью), так и функциональностью навигационной (по виду полосы прокрутки пользователь может определить, в какой области документа он находится).

За время существования полосы прокрутки уже предпринимались попытки улучшить ее эргономические характеристики за счет усиления навигационной функции. Но большинство из этих предложений так и осталось на уровне нереализованных теорий. Например, Бен Шнейдерман в своей книге *Designing of the User Interface* [2] предлагает

несколько вариантов видоизменения стандартной полосы прокрутки, направленных на получение дополнительных выгод от ее использования:



Предложенные Шнейдерманом варианты:

- номер страницы выводится на самом бегунке;
- бегунок пропорционален размеру просматриваемой области;
- на полосе прокрутки показываются места закладок;
- на полосе прокрутки показываются разделы документа;
- полоса прокрутки с жестко выбранными возможными позициями бегунка.

Ни одно из предложений Шнейдермана так и не было использовано в современных приложениях, за исключением пропорционального бегунка, несмотря на очевидное стремление разработчиков предоставить пользователям инструменты, по сути решающие те же задачи, что и функционально дополненная полоса прокрутки.

Таким образом, нахождение альтернативных средств прокрутки вновь является актуальной задачей.

В данной работе представлен прототип интерфейсного решения, использующий диагональную прокрутку в графическом интерфейсе мобильных устройств. Главным элементом предлагаемого механизма являются две кнопки, расположенные соответственно в левом верхнем и правом нижнем углах рабочей области вывода и совпадающие по размеру с пиктограммами. Таким образом достигается увеличение размера, актуальное как с т.з. закона Фиттса, так и для пальцевого управления, при экономном использовании пространства экрана.

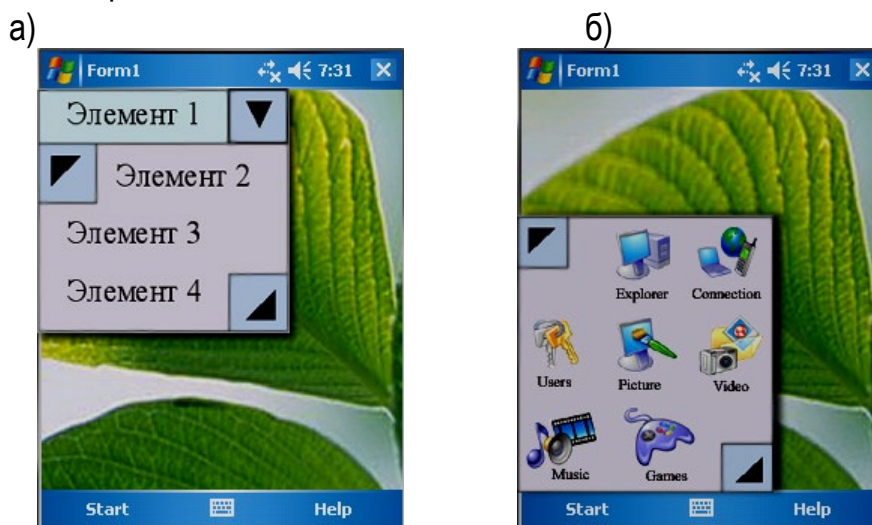


Рисунок 2 – Диагональная прокрутка: а) в выпадающем списке; б) в контейнере пиктограмм

Нажатием на одну из кнопок выполняется прокрутка пиктограмм вдоль главной диагонали области вывода, т.е. в благоприятном для интуитивного восприятия направлении [1]. Алгоритм прокрутки подразумевает деление набора пиктограмм на секции, границы которых ортогональны главной диагонали окна. Скрытие и показ пиктограмм выполняется посекционно.

Для воспроизведения присущей стандартной полосе прокрутки информационной функции в предлагаемом интерфейсном решении кнопка скроллинга меняет оттенок цвета в зависимости от количества пиктограмм, находящихся в соответствующей ей скрытой зоне контейнера, становясь более темной по мере «заполнения» скрытой зоны. Тем самым выполняются основные принципы отображения информации, разработанные Эдвардом Тафтом.

Необходимо отметить, что всегда необходимо искать компромисс между сложностью экрана и сложностью навигации. Этот компромисс зависит от скорости и простоты работы навигации и от структуры данных.

Литература

1. Раскин, Дж. Интерфейс: новые направления в проектировании компьютерных систем / Дж. Раскин – СПб.: Символ-Плюс, 2003.
2. Shneiderman B., Plaisant C. Designing of the User Interface: Strategies for Effective Human-Computer Interaction. Addison Wesley, 2004.

УДК 621.374:681.511

АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ ГИДРОДИНАМИКИ ЦЕМЕНТИРОВАНИЯ НЕФТЯНЫХ СКВАЖИН

Фарберов А.Г., Трохова Т.А.

УО «Гомельский государственный технический университет имени П. О. Сухого», г. Гомель

Информационные технологии все глубже проникают в различные отрасли промышленности Республики Беларусь. Однако наиболее пристальное внимание к этому процессу уделяется в тех отраслях промышленности, которые, в свете последних событий, должны обеспечить энергетическую безопасность страны. Одной из таких отраслей является нефтедобыча. Здесь, как и во многих других отраслях, основной задачей, стоящей перед специалистами в сфере информационных технологий, является автоматизация трудоемких инженерных работ.

Автоматизация процесса строительства нефтяных скважин ведется в нескольких направлениях, основными из которых являются следующие:

- планирование работ по строительству скважин;
- оперативный контроль за этим строительством;
- инженерное проектирование строительства скважин.

Раздел в проектировании строительства нефтяных скважин, для которого разрабатывается программный комплекс, называется «Расчет параметров гидродинамики цементирования обсадных колонн». Применение новых информационных технологий в данной сфере дает возможность разработки адекватных моделей гидродинамики цементирования обсадных колонн нефтяных скважин.

Задача автоматизации расчетов и моделирования гидродинамики процессов в обсадной колонне и затрубном пространстве при цементировании на сегодняшний день является актуальной, т.к. позволяет:

- значительно повысить качество проектирования;
- сократить время проведения расчетов.

Структура системы автоматизации проектирования гидродинамики цементирования представлена несколькими взаимосвязанными блоками, каждый из которых поддерживает свой режим работы. Основными блоками являются:

- блок ввода исходных данных;
- блок расчета параметров цементирования;
- блок расчета количества и режимов работы цементировочных агрегатов и цементосмесительных машин;
- блок расчета гидродинамики процесса цементирования;
- блок графического моделирования.

Существует несколько способов цементирования скважин. Они различаются схемой подачи тампонажного раствора в затрубное пространство и особенностями используемых приспособлений. Если через башмак обсадной колонны в затрубное пространство продавливают весь тампонажный раствор, то способ называется одноступенчатым (одноцикловым) цементированием. Если обсадная колонна на разных уровнях оснащена дополнительными приспособлениями (заливочными муфтами), позволяющими подавать тампонажный раствор в затрубное пространство поинтервально на разной глубине, то способ цементирования называется многоступенчатым (многочикловым).

Наиболее простым и распространенным в промышленных масштабах является одноступенчатое цементирование по прямой схеме.

Общий порядок процесса цементирования следующий. После бурения скважина заполнена промывочной жидкостью. В скважину опускается обсадная колонна. Далее для раздела промывочной жидкости от цемента (тмпонажного раствора), т.е. предотвращения их смешивания, в колонну нагнетают буферную жидкость. После завершения этой операции начинают закачку требуемого объема тампонажного раствора в колонну, а далее для выталкивания его из пространства колонны и заполнения затрубного пространства нагнетают продавочную жидкость.

Рассмотрим порядок функционирования разработанной программы. При работе в режиме расчета параметров цементирования реализуется алгоритм проектного расчета с получением таких основных расчетных результатов, как объем цементного раствора, масса сухого цемента, объем буферной жидкости и воды для затворения раствора, объем продавочной жидкости и т.д. В настоящем проекте эти расчеты отлажены и совпадают с эталонным примером.

Блок расчета количества и режимов работы цементировочных агрегатов и цементосмесительных машин позволяет на основе расчетных параметров цементирования подобрать количество агрегатов так, чтобы соблюдались все технологические требования при строительстве скважин. Одним из основных критериев выбора является время цементирования.

С использованием средства разработки приложений для операционной системы Windows (среда программирования Borland Delphi 7, язык программирования Visual Pascal) была создана прикладная программа, моделирующая процесс цементирования скважин.

Исходными данными для реализации модели являются глубина скважины, диаметр скважины, диаметр колонны.

Для каждой жидкости, закачиваемой в скважину (кроме промывочной), пользователь выбирает цвет на рисунке. Цвет также может быть принят по умолчанию.

При построении анимационной модели эскиз скважины с обсадными колоннами программно масштабируется по глубине и ширине различными масштабами, т.к. диапазон изменения глубины скважины в среднем варьируется от 1000 до 6000 метров, а диапазон изменения диаметра скважины от 0,15 до 0,5 метра. Программа выполняет масштабирование автоматически.

Кроме того, пользователь вводит масштаб времени — это то количество минут реального цементирования, приходящееся за одну минуту времени работы программы, например, общее время цементирования 40 минут (общее время закачки всех жидкостей), то при задании масштаба времени 1:10 минутам, программа будет работать 4 минуты. Масштаб времени является исходным данным и выбирается пользователем и может изменяться в процессе моделирования, ускоряя или замедляя процесс. Внешний вид сконструированного интерфейса пользователя представлен на рисунке 1.

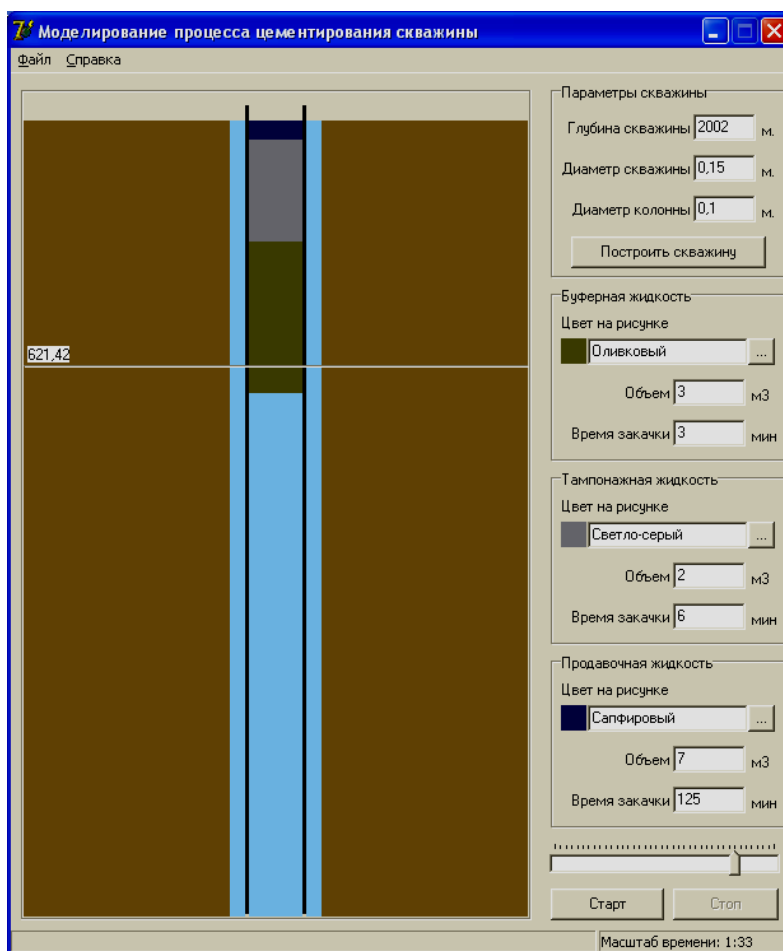


Рисунок 1 — Внешний вид программы в процессе моделирования

Разработанная система отлажена на тестовом примере, предоставленном институтом БелНИПИ Республиканского унитарного предприятия «Производственное объединение «Белоруснефть»».

МОДЕЛИРОВАНИЕ УРОВНЕЙ ПОДЗЕМНЫХ ВОД ПРИ РАЗРАБОТКЕ ТОРФЯНЫХ МЕСТОРОЖДЕНИЙ (НА ПРИМЕРЕ «ГРИЧИНО-СТАРОБИНСКОГО ТОРФЯНОГО МЕСТОРОЖДЕНИЯ»)

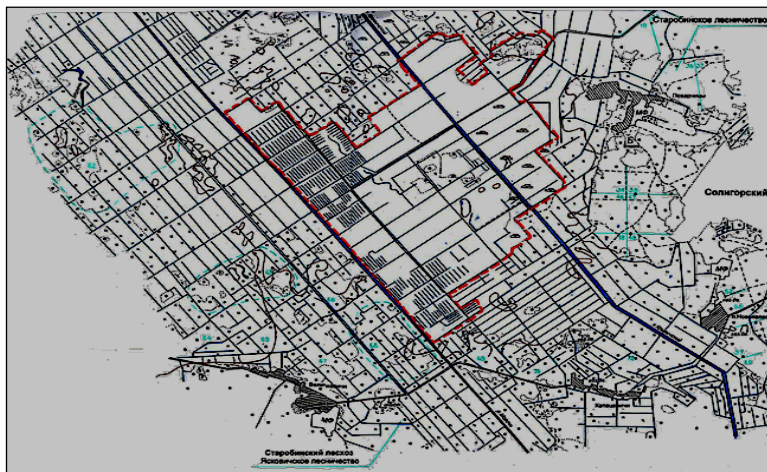
Хинич А.И.

УО «Брестский государственный технический университет», г. Брест

В настоящее время для разработки любого природного месторождения требуется предварительная экологическая экспертиза. В результате анализа методов и способов оценки изменений водного режима разрабатываемых торфяных месторождений и прилегающих территорий наиболее приемлемыми являются методы математического моделирования. Ввиду большого разнообразия природно-климатических условий для каждого конкретного объекта типовые расчетные схемы должны корректироваться с учетом индивидуальных особенностей разрабатываемого торфяного месторождения. Основными требованиями при выборе участка для наблюдений за режимом грунтовых вод является его типичность для данной территории.

Для организации участка необходимо иметь следующие материалы: план расположения осушительной сети торфяного месторождения и сооружений, данные о глубине торфяной залежи и места установки наблюдательных колодцев, характере грунта ниже торфяной залежи, ботаническом составе, степени разложения и коэффициенте фильтрации торфа, типе водного питания для данного участка.

Для наблюдения за уровнями грунтовых вод при разработке торфяных месторождений было выбрано «Гричино-Старобинское месторождение».



**Рисунок 1 – «Гричино-Старобинское торфяное месторождение»
Солигорского района
Минской области**

Месторождение находится в обширной заболоченной пойме р. Морочь. Ширина массива – до 10 км., длина около 20 км. В геоморфологическом отношении рельеф представляет собой пойменно-озерную равнину, сложенную торфом и разбросанными по всей территории невысокими суходольными возвышенностями. Торфяная залежь сложена торфами низинного типа, из которых преобладает осоковый – 33,4 %, древесно-тростниковый – 24,8 %, тростниковый – 22,8 %, древесно-осоковый – 7,4 %. Положение грунтовых вод находится в прямой зависимости от количества выпадающих атмосферных осадков и от уровня воды в каналах. Ложе месторождения торфа, а также внутренние суходольные участки, сложены песками мелкими и средней крупности, с маломощными прослойками супесей.

Фрезерные поля осушаются картовыми каналами, располагаемыми через 40 метров, которые соединены в плане под прямым углом с валовыми каналами. Расстояние между валовыми каналами принято 500 метров.

Глубина пласта для расчетов разработки 0,8 м. при общей средней глубине 1,76 м. Средние значения технических свойств торфа по всему участку характеризуются следующими показателями:

- степень разложения – 35% ;
- влажность – 81,1%;
- зольность – 14,1%.

Расчетная промышленная площадь составляет 1256,6 га. в границе промышленной глубины залежи торфа 0,8 м. В таблице 1 приведены параметры кривой распределения стока реки-аналога Морочь – с. Мацкевичи.

Таблица 1 – Параметры кривой распределения характерного стока реки – аналога Морочь – с. Мацкевичи

Характеристика стока	Собственный ряд					Приведенный ряд				
	П	м³/с	л/с.км²	C _v	C _s	П	м³/с	л/с.км²	C _v	C _s
Годовой сток	37	1,37	4,38	0,30	0,78	55	1,46	4,66	0,30	0,84
Весеннее половодье	37	30,6	97,8	0,79	1,15	55	35,3	113,0	0,77	1,26
Слой стока весеннего половодья	33	-	-	0,54	0,47	51	-	-	0,48	0,34
Дождевой паводок	37	5,71	18,2	0,81	2,11	-	-	-	-	-
Минимальный летний сток	38	0,24	0,77	0,41	0,45	-	-	-	-	-

Моделирование уровней подземных вод возможно по следующим расчетным схемам.

Для потока грунтовых вод, при котором струйки воды перемещаются в параллельных вертикальных плоскостях, удобно рассматривать расход потока на единицу его ширины [1]

$$q = Q / B, \tag{1}$$

где q — единичный расход потока (расход, через 1 м ширины потока), м³/сут; Q – общий расход потока, м³/сут; B – ширина потока, м.

Согласно закону Дарси, единичный расход потока в пределах рассматриваемого участка равен

$$q = k \cdot h_{cp} \cdot I_{cp}, \tag{2}$$

где k – коэффициент фильтрации водоносного пласта; h_{cp} – средняя мощность потока; I_{cp} – средний напорный градиент потока.

В случае совпадения уклона водоносного пласта с направлением движения грунтовых вод (рис. 2 и 3) уравнение кривой депрессии имеет вид:

$$\frac{i \cdot \ell}{h_0} = \varphi(\eta_2) - \varphi(\eta_1), \tag{3}$$

где i – уклон поверхности водоупора в долях единицы; ℓ – расстояние между сечениями, м; h_0 – приведенная мощность потока, м, при которой расход его равен $q = k h_0 i$, что отвечает равномерному движению воды с уклоном зеркала воды i ; $\varphi(\eta_2)$ – значение функции $\varphi(\eta)$ при относительной мощности потока $\eta = h_2 / h_0$; $\varphi(\eta_1)$ – значение той же функции при $\eta = h_1 / h_0$, для кривой подпора $\varphi(\eta) = (\eta) + \ell n(\eta - 1)$, для кривой спада $\varphi(\eta) = \eta + \ell n(1 - \eta)$.

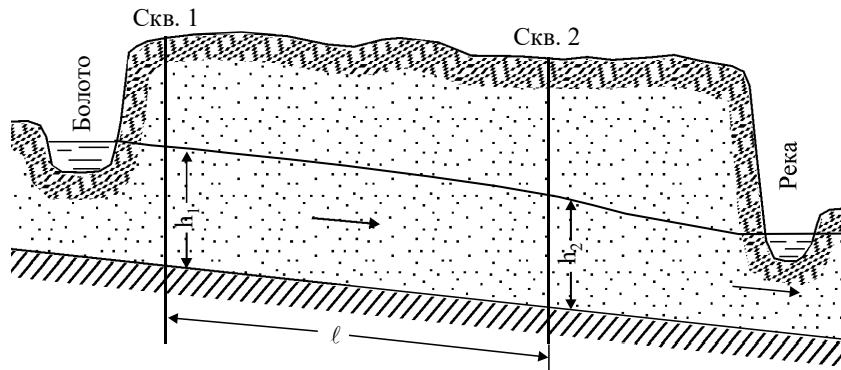


Рисунок 2 – Кривая подпора при прямом уклоне основания водоносного пласта

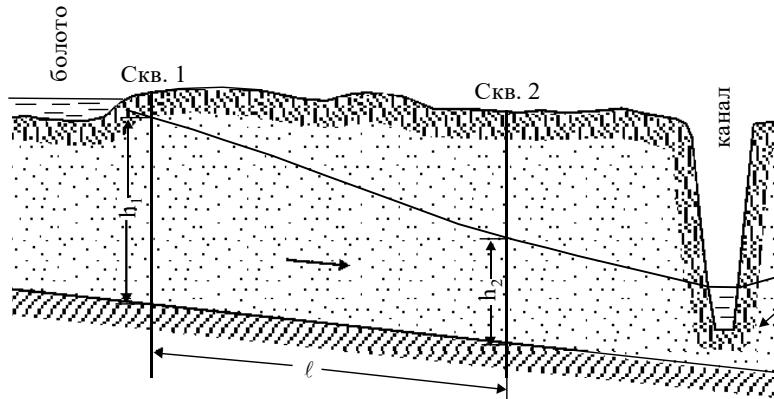


Рисунок 3 – Кривая спада при прямом уклоне основания водоносного пласта

Значения функции $\varphi(\eta)$ в зависимости от значений η находятся по специальным таблицам. При $\eta > 1$ наблюдается кривая подпора (рис. 2), при $\eta < 1$ – кривая спада (рис. 3). Кривая подпора характеризуется увеличением мощностей потока по направлению движения, кривая спада – уменьшением мощностей в том же направлении.

При обратном уклоне водоупорного ложа потока, при котором направление потока противоположно падению водоупора, возможна лишь кривая спада. Для нее характерно убывание мощности потока по течению $\eta = h / h_0 < 1$ (рис. 4).

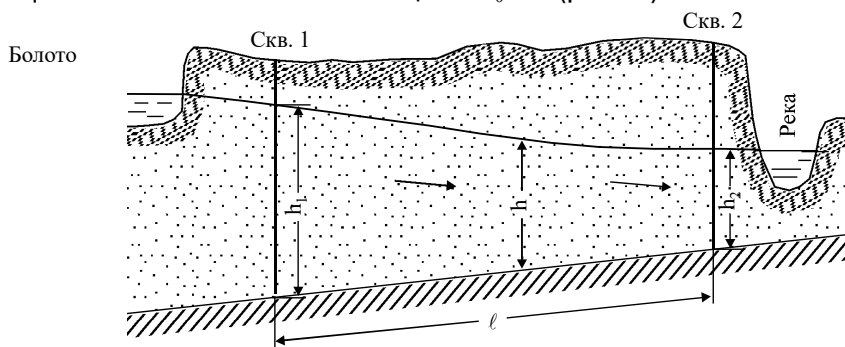


Рисунок 4 – Кривая спада при уклоне основания водоносного пласта

Для кривой депрессии в этом случае действительно уравнение:

$$\frac{i \cdot l}{h'_0} = \psi(\eta_2) - \psi(\eta_1), \tag{4}$$

где h'_0 – приведенная мощность потока; $\psi(\eta_1)$ и $\psi(\eta_2)$ – значения функции $\psi(\eta) = -\eta + \ln(1 + \eta)$ соответственно для значений $\eta_1 = h_1 / h'_0$ и $\eta_2 = h_2 / h'_0$.

При движении грунтовых вод в неоднородном наклонном пласте сложного строения, водопроницаемость которого изменяется в горизонтальном и вертикальном направлениях без какой-либо ясной закономерности (общий случай разнородности) расход потока определяется из уравнения:

$$q = \frac{k_1 \cdot h_1 + k_2 \cdot h_2}{2} \cdot \frac{H_1 - H_2}{\ell}, \quad (5)$$

где k_1 и k_2 — средние коэффициенты фильтрации пласта в верхнем и нижнем сечениях потока; h_1 и h_2 мощности потока в тех же сечениях.

Средний коэффициент фильтрации для данного сечения определяется по данным послойных определений водопроницаемости лабораторным способом. Зная послойные коэффициенты фильтрации k_1, k_2, \dots, k_n и соответствующие мощности отдельных слоев $h_1, h_2, h_3, \dots, h_n$, вычисляют средневзвешенное значение коэффициента фильтрации для всего сечения по формуле:

$$k = \frac{k_1 \cdot h_1 + k_2 \cdot h_2 + k_3 \cdot h_3 + \dots + k_n \cdot h_n}{h_1 + h_2 + h_3 + \dots + h_n} = \frac{\sum kh}{\sum h}. \quad (6)$$

Для водоносного пласта, состоящего из двух горизонтально залегающих слоев, действительно уравнение:

$$q = k_1 \cdot M \cdot \frac{h_1 - h_2}{\ell} + k_2 \cdot \frac{h_1 + h_2}{2} \cdot \frac{h_1 - h_2}{\ell}, \quad (7)$$

где k_1 — коэффициент фильтрации нижнего слоя; k_2 — то же для верхнего слоя; M — мощность нижнего слоя; h_1 — уровень грунтовых вод в верхнем сечении потока, считая от кровли нижнего слоя; h_2 — то же в нижнем сечении потока; ℓ — расстояние между выбранными сечениями.

При резкой смене водопроницаемости водоносного пласта в горизонтальном направлении, что наблюдается, например, при переходе от водораздельного массива к аллювиальной долине (на террасе), может быть использована следующая формула Каменского:

$$q = \frac{h_1^2 - h_2^2}{2 \cdot \left(\frac{\ell_1}{k_1} + \frac{\ell_2}{k_2} \right)}. \quad (8)$$

С помощью математических моделей можно в той или иной степени оценить величину антропогенных воздействий на экосистему.

Литература

1. Бурлибаев, М.Ж. Гидрологические измерения и гидрогеологические расчеты для водохозяйственных целей / М.Ж. Бурлибаев, А.А. Волчек, М.Ю. Калинин – Алматы: Издательство «Каганат», 2004. – 358 с.

ОПТИМИЗАЦИЯ И КООРДИНАЦИЯ УПРАВЛЕНИЯ СВЕТОФОРАМИ

Шуть В.Н., Войцехович О.Ю., Анфилец С.В.

УО «Брестский государственный технический университет», г. Брест

1. Автоматизированные системы управления дорожным движением (АСУДД).

С точки зрения системного подхода, все методы управления дорожным движением можно разделить на методы, действующие в реальном времени (on-line методы) и вне его (off-line методы).

К первой группе относятся ручное регулирование на перекрестке, а также алгоритмы автоматизированного управления, основанные на получении информации от датчиков транспортных потоков. Такие алгоритмы широко применяются в используемых за рубежом АСУДД.

Ко второй группе относятся практически все методы принудительного распределения транспортных потоков, реализуемые посредством дорожных знаков и дорожной разметки. Сюда же следует отнести и светофорную сигнализацию с неизменными в суточном цикле параметрами регулирования.

Каждую АСУДД относят к одному из четырех поколений.

I поколение. Расчет управляющих параметров и ввод их в АСУДД выполняются вручную.

II поколение. Расчет управляющих параметров автоматизирован, ввод их в АСУДД выполняются вручную.

III поколение. Расчет управляющих параметров и ввод их в АСУДД автоматизированы. Управление прогнозируется.

IV поколение. Управление в реальном времени.

АСУДД, установленные в большинстве городов Беларуси, относятся к 1-му поколению. Была поставлена задача перевести уже существующую АСУДД в г. Бресте из 1-го поколения во 2-е.

2. Параметры для оптимизации. Существенные параметры работы сигнализированного перекрестка - или сети перекрестков - время цикла, зеленый раскол и смещение, связанное с общим циклом времени.

Количество последовательного времени зеленого сигнала, которое может быть распределено по фазам перекрестков, зависит от времени цикла. Уменьшение задержек транспортных средств ведет к короткой длине цикла, в то время как уменьшение остановок означает продолжительную длину цикла.

Время цикла перекрестка также влияет на безопасность сети т.к.:

1) если красный свет горит слишком долго, некоторые автомобилисты могут начать игнорировать сигналы;

2) слишком частый цикл сигналов увеличивает риск столкновений.

Зеленый раскол - это отношение времени зеленого света к времени цикла для фаз перекрестка. Зависит от ожидаемого количества транспорта от каждой дороги, смежной с перекрестком.

Смещение позволяет создать зеленые волны для перемещения автотранспорта от одного перекрестка к другому в конкретном темпе. Параметр уместен при оптимизации сигнала для главных дорог.

Почти все методы оптимизируют некоторые или все эти три параметра. Реже оптимизируется такой параметр, как фазовая последовательность, то есть порядок красных и зеленых сигналов в цикле. Это главным образом уместно при предоставлении приоритетов некоторым видам транспорта, например, общественному транспорту.

3. Зеленая волна. Один из способов уменьшить количество остановок в транспортном потоке – организовать переключение сигналов так, чтобы наименьшим образом препятствовать транспортному потоку.

Зеленая волна – это специально созданное явление, в котором последовательность светофоров (3 или более) скоординирована так, что позволяют транспортному потоку двигаться непрерывно через некоторое число перекрестков в главном направлении.

При умеренной интенсивности движения и относительно небольших расстояниях между перекрестками (~500 метров) образуется групповая форма движения, обусловленная чередованием фаз светофорной сигнализации. Т.о. координированное управление формирует случайно возникающие группы ТС в динамически однородные группы автомобилей (пачки). Перекрестки становятся взаимозависимыми по управлению. Возникает необходимость в координации сигналов путем использования периодического, группового характера транспортного потока. Это увеличивает пропускную способность дорог, уменьшает шум и затраты энергии.

В некоторых случаях зеленые волны могут взаимодействовать друг с другом, но это повышает их сложность и ведет к нестабильности, поэтому только дороги и направления с наиболее интенсивным движением получают такую привилегию.

Зеленые волны в обоих направлениях могут быть созданы с помощью рекомендации различных скоростей для каждого направления, иначе транспорт, подходящий к перекрестку с одного направления, может достичь светофора быстрее, чем транспорт с другого направления, если расстояние от предыдущего светофора не равно в обоих направлениях.

4. Методы регулирования светофоров для автономных систем. Методы регулирования светофоров для систем с фиксированным временем глобально делятся на два класса.

Первый и исторически более старый состоит из методов, которые максимизируют полосу пропускания и продвижение вперед. (Полоса пропускания – это временной промежуток, когда водитель в транспортном потоке может проехать два или более светофора, но без остановок). Вторая группа содержит методы, которые стремятся минимизировать задержку или другие негативные показатели.

В данной работе предпочтение отдано методам с поиском максимальной полосы пропускания. Такие методы используют относительно маленькую входную информацию, основными требованиями являются указания уличной геометрии, скорости движения и зеленых расколов, являются надежными в эксплуатации.

5. Алгоритм решения задачи координации светофоров. Вывод оптимальной функции координации происходит с помощью рисунка 1. Разница во времени от точки (А) до (В) выражается двумя различными путями: сначала используя переменные, определенные с помощью величин выходящего потока, затем используя величины входящего потока.

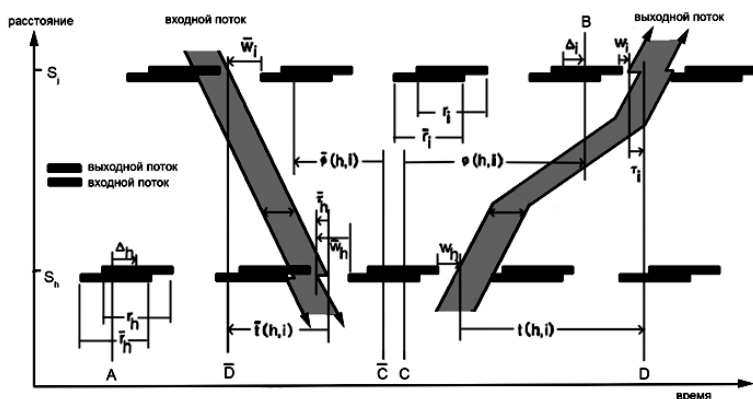


Рисунок 1 – Пространственно-временная диаграмма, показывающая зеленые волны

На рис. 1 входные и выходные зеленые волны проходят через светофоры S_i и S_h . Величины со штрихами относятся к входящим направлениям, а без – к исходящим. Красные сигналы в исходящих потоках нарисованы сплошной полосой, входящие - пунктирной. В общем случае входящие и исходящие сигналы не требуется различать.

Путем математических преобразований и физических ограничений выводится линейная целевая функция (1) для настройки артериальных светофоров.

$$b + k\bar{b} \rightarrow \max, \quad (1)$$

где b (\bar{b}) – исходящая (входящая) полоса пропускания (в циклах);

k – отношение входящей полосы пропускания к исходящей.

Математическая модель включает $(11n-10)$ ограничений и $(4n + 1)$ вещественных переменных, до $2n$ переменных, принимающих значения 0 или 1, $n-1$ неограниченных целых переменных (n – количество перекрестков).

Таким образом, оптимизация сигналов регулирования транспортного движения является многокритериальной задачей. Результатом работы является портативная, автономная, написанная на языках C++ и Matlab компьютерная программа для настройки светофоров на крупной центральной дороге для достижения максимальной полосы пропускания. К специфическим особенностям программы можно отнести:

1) автоматически выбираемое время цикла из предложенного диапазона, 2) возможность модификации скорости в допустимых пределах 3) возможность введения времени рассеивания очереди для вторичного потока, накопившегося в течение красного, 4) зеленые расколы могут быть даны или, в качестве альтернативы, по заданным потокам и пропускным способностям расколы подсчитываются с использованием теории Вебстера и т.д.

Программа выдает время цикла, смещения, скорости и порядок фаз поворота налево, чтобы максимизировать взвешенную комбинацию полос пропускания.

Заключение. Цель координированных сигналов – обеспечение полосы пропускания в оба направления настолько широкой, насколько это возможно. Разработанная компьютерная программа рассчитывает время цикла, смещения, скорости и пр., чтобы максимизировать взвешенную комбинацию полос пропускания, при наличии расстояний, интенсивности и пропускной способности, времени очищения очереди и др.

Настройки, полученные от программы, необходимо проверить в программе моделирования транспорта. Кроме того, есть возможность расширить существующую целевую функцию, а в более далекой перспективе перейти к адаптивному управлению.

Литература

1. Врубель, Ю.А. Организация дорожного движения / Ю.А. Врубель – Минск, 1996. – Ч. 2
2. Кременец, Ю.А. Технические средства регулирования дорожного движения / Ю.А. Кременец, М.П. Печерский – Москва: Транспорт, 1981.
3. Dotoli, M., Fanti, M.P. and Meloni C. A signal timing plan formulation for urban traffic control. Control Engineering Practice, 14:1297 – 1311, 2006.
4. Warberg, A. Green wave traffic optimization. Technical report, Technical University of Denmark, 2007.
5. Webster, F.V. Traffic signal setting. Great Britain Department of Scientific and Industrial Research, 1958.
6. Ying, J.Q., Lu, H. and Shi, J. An algorithm for local continuous optimization of traffic signals. European Journal of Operational Research, 2007.

АНАЛИТИЧЕСКИЕ И ЧИСЛЕННЫЕ МЕТОДЫ ИССЛЕДОВАНИЙ В МАТЕМАТИКЕ И ИХ ПРИЛОЖЕНИЯ

УДК 519.6+517.983.54

СХОДИМОСТЬ МЕТОДА ИТЕРАЦИЙ ЯВНОГО ТИПА В ЭНЕРГЕТИЧЕСКОЙ НОРМЕ ГИЛЬБЕРТОВА ПРОСТРАНСТВА

Альциванович Л.М.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

В действительном гильбертовом пространстве H решается уравнение первого рода

$$Ax = y, \quad (1)$$

где A – ограниченный положительный самосопряженный оператор, для которого нуль не является собственным значением. Однако нуль принадлежит спектру оператора A и, следовательно, задача некорректна.

Исследуем сходимость к решению уравнения (1) методов

$$x_{n+1} = (E - \alpha A^2)x_n + \alpha Ay, \quad x_0 = 0, \quad (2)$$

$$x_{n+1,\delta} = (E - \alpha A^2)x_{n,\delta} + \alpha Ay_\delta, \quad x_{0,\delta} = 0 \quad (3)$$

в энергетической норме $\|x\|_A = \sqrt{(Ax, x)}$. Покажем, что использование энергетической нормы позволяет получить оценки без предположения истокорпредставимости точного решения уравнения (1), т. е. что $x = A^s z$, $s > 0$. Получим оценки погрешности для метода (3) в энергетической норме. Справедлива

Теорема 1. *При условии*

$$0 < \alpha \leq \frac{5}{4\|A\|^2}, \quad (4)$$

итерационный процесс (3) сходится в энергетической норме гильбертова пространства, если выбирать число итераций $n(\delta)$ из условий $n^{\frac{1}{4}}\delta \rightarrow 0, n \rightarrow \infty, \delta \rightarrow 0$.

Доказательство.

Нетрудно показать, что $x - x_n = A^{-1}(E - \alpha A^2)^n y = (E - \alpha A^2)^n x$. Воспользовавшись определением энергетической нормы $\|x\|_A = \sqrt{(Ax, x)}$ и интегральным представлением самосопряженного оператора, запишем

$$\|x - x_n\|_A^2 = \left(A(E - \alpha A^2)^n x, (E - \alpha A^2)^n x \right) = \int_0^{\|A\|} \lambda (1 - \alpha \lambda^2)^{2n} d(E_\lambda x, x). \text{ Для положительной}$$

подынтегральной функции $f(\lambda) = \lambda (1 - \alpha \lambda^2)^{2n}$ при условии (4) справедлива оценка

$$\max_{\lambda \in [0, M]} |f(\lambda)| \leq (4n\alpha e)^{-1/2}. \text{ Следовательно, } \|x - x_n\|_A \leq (4n\alpha e)^{-1/4} \|x\|.$$

Отсюда получим $\|x - x_n\|_A \rightarrow 0$, $n \rightarrow \infty$. Оценим $\|x_n - x_{n,\delta}\|_A$.

Имеем $\|x_n - x_{n,\delta}\|_A^2 = \int_0^{\|A\|} \lambda^{-1} \left[1 - (1 - \alpha\lambda^2)^n \right]^2 d(E_\lambda(y - y_\delta), y - y_\delta)$. Оценим сверху

подынтегральную функцию $g_n(\lambda) = \lambda^{-1} \left[1 - (1 - \alpha\lambda^2)^n \right]^2 \geq 0$ при условии (4).

Имеем $g_n(\lambda) = \lambda^{-1} \left[1 - (1 - \alpha\lambda^2)^n \right] \left[1 - (1 - \alpha\lambda^2)^n \right]$.

Так как $\lambda^{-1} \left[1 - (1 - \alpha\lambda^2)^n \right] \leq \left(\frac{5}{4}\right)^{1/2} 2(n\alpha)^{1/2}$ при $n \geq 1$, $\lambda^{-1} \left[1 - (1 - \alpha\lambda^2)^n \right] \leq 2(n\alpha)^{1/2}$

при $n \geq 2$ и $\left| 1 - (1 - \alpha\lambda^2)^n \right| \leq 2$, то $g_n(\lambda) \leq \left(\frac{5}{4}\right)^{1/2} 4(n\alpha)^{1/2}$ при $n \geq 1$, $g_n(\lambda) \leq 4(n\alpha)^{1/2}$

при $n \geq 2$. Следовательно, при условии (4) $\|x_n - x_{n,\delta}\|_A \leq \left(\frac{5}{4}\right)^{1/2} 2n^{1/4} \alpha^{1/4} \delta$, $n \geq 1$,

$\|x_n - x_{n,\delta}\|_A \leq 2n^{1/4} \alpha^{1/4} \delta$, $n \geq 2$.

Поскольку $\|x_n - x_{n,\delta}\|_A \leq \|x - x_n\|_A + \|x_n - x_{n,\delta}\|_A \leq \|x - x_n\|_A + 2n^{1/4} \alpha^{1/4} \delta$, $n \geq 2$ и $\|x - x_n\|_A \rightarrow 0$, $n \rightarrow \infty$, то для сходимости $\|x - x_{n,\delta}\|_A \rightarrow 0$, $n \rightarrow \infty$, достаточно, что-

бы $n^{1/4} \delta \rightarrow 0$, $n \rightarrow \infty$, $\delta \rightarrow 0$. Таким образом, если в процессе (3) выбрать число итераций $n = n(\delta)$, зависящим от δ так, чтобы $n^{1/4} \delta \rightarrow 0$, $n \rightarrow \infty$, $\delta \rightarrow 0$, то получим регуляризованный метод, обеспечивающий сходимость к точному решению в энергетической норме. Теорема 1 доказана.

Запишем общую оценку погрешности метода (3) в энергетической норме

$$\|x - x_{n,\delta}\|_A \leq \left(\frac{5}{4}\right)^{1/2} 2n^{1/4} \alpha^{1/4} \delta + (4n\alpha e)^{-1/4} \|x\|, \quad n \geq 1, \quad (5)$$

$$\|x - x_{n,\delta}\|_A \leq 2n^{1/4} \alpha^{1/4} \delta + (4n\alpha e)^{-1/4} \|x\|, \quad n \geq 2. \quad (6)$$

Полученные результаты объединим в теорему

Теорема 2. Для итерационного процесса (3) при условии (4) справедливы оценки погрешности (5) и (6).

Оптимизируем по n полученную оценку (6). Приравняв к нулю производную по n от правой части неравенства (6), имеем

$$n_{\text{opt}} = 2^{-3} \alpha^{-1} e^{-\frac{1}{2}} \|x\|^2 \delta^{-2}. \quad (7)$$

Подставим $n_{\text{ндо}}$ в оценку (6), получим оптимальную оценку погрешности

$$\|x - x_{n,\delta}\|_A^{\text{ндо}} \leq 2^{\frac{5}{4}} e^{-\frac{1}{8}\frac{1}{\delta^2}} \|x\|^{\frac{1}{2}}, \quad n \geq 2. \quad (8)$$

Итак, доказана

Теорема 3. В условиях предыдущей теоремы оптимальная оценка погрешности в энергетической норме для метода (3) имеет вид (8) и получается при $n_{\text{ндо}}$ из (7).

Таким образом, энергетическая норма как бы заменяет истокорпредставимость точного решения уравнения (1) степени $s = \frac{1}{2}$. Её использование позволило получить априорные оценки погрешности и априорный момент останова без дополнительного требования на гладкость точного решения – его истокообразную представимость. Следовательно, энергетическая норма позволяет сделать итерационный процесс (3) эффективным и тогда, когда нет сведений об истокопредставимости точного решения.

Важен вопрос о том, когда из сходимости в энергетической норме следует сходимость в обычной норме гильбертова пространства H . Эти условия дает

Теорема 4. Если выполнены условия 1) $E_\varepsilon x_{n,\delta} = 0$, 2) $E_\varepsilon x = 0$, где $E_\varepsilon = \int_0^\varepsilon dE_\lambda$, ε – фиксированное положительное число ($0 < \varepsilon < \|A\|$), то из сходимости $x_{n,\delta}$ к x в энергетической норме следует сходимость в обычной норме гильбертова пространства.

УДК 517.948

ОБ ЭФФЕКТИВНОСТИ ОДНОШАГОВЫХ И МНОГОШАГОВЫХ МЕТОДОВ РЕШЕНИЯ НЕЛИНЕЙНЫХ СИСТЕМ

Артюшеня А.А.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

Для решения нелинейной системы

$$f(x) = 0, f(D \in R^n \rightarrow R^n), f \in C_D^{(2)} \quad (1)$$

применяются одношаговые, двухшаговые и трехшаговые методы.

В данной работе для решения модельной системы вида [2]:

$$\begin{cases} 2x_1 + x_2 + \dots + x_{n-2} + x_{n-1} + x_n = n + 1, \\ x_1 + 2x_2 + \dots + x_{n-2} + x_{n-1} + x_n = n + 1, \\ x_1 + x_2 + \dots + 2x_{n-2} + x_{n-1} + x_n = n + 1, \\ x_1 x_2 \dots x_{n-2} x_{n-1} x_n = 1, \\ \sin^2 x_1 + \cos^3 x_n = \sin^2 1 + \cos^3 1 \end{cases} \quad (2)$$

применялись одношаговые, двухшаговые и трехшаговые методы.

Алгоритм решения нелинейной системы следующий:

Шаг 1: решается система линейных алгебраических уравнений:

$$f'(x_n)\Delta x_n = -f(x_n), \quad n = 0, 1, 2, \dots \quad (3)$$

где $f'(x_n)$ - матрица Якоби, Δx_n - искомый вектор.

Шаг 2: вычисляется вектор x_{n+1} в соответствии с каждым из методов:

$$x_{n+1} = x_n + \sqrt{\beta_n} \Delta x_n, \quad n = 0, 1, 2, \dots, \quad \beta_0 \in [10^{-3}, 10^{-1}]. \quad (4)$$

$$x_{n+1} = x_n + \sqrt[3]{\beta_n} \Delta x_n, \quad n = 0, 1, 2, \dots, \quad \beta_0 \in [10^{-3}, 10^{-1}]. \quad (5)$$

$$x_{n+1} = x_n + \beta_n \Delta x_n, \quad n = 0, 1, 2, \dots, \quad \beta_0 \in [10^{-3}, 10^{-1}]. \quad (6)$$

$$x_{n+1} = x_n + \sqrt[4]{\beta_n} \Delta x_n, \quad n = 0, 1, 2, \dots, \quad \beta_0 \in [10^{-3}, 10^{-1}]. \quad (7)$$

Шаг 3: проверяется условие:

$$\|f(x_{n+1})\| < \varepsilon, \quad \text{где } \|f(x_n)\| = \sqrt{\sum_{i=1}^n f_i^2(x_n)}. \quad (8)$$

Если условие (8) выполняется, то x_{n+1} – принимается за приближенное решение системы (1). В противном случае производится пересчет β_{n+1} , для чего проверяется условие: если

$$\|f(x_{n+1})\| < \|f(x_{n+1})\|,$$

то $\beta_{n+1} = 1$, иначе β_{n+1} находится по одной из формул (9)-(13), определенной для каждого метода, и осуществляем переход на шаг 1.

Будем рассматривать трехшаговые методы, где β_n просчитывается по формулам (9)-(11):

$$\beta_{n+1} = \min \left(1, \frac{\gamma_n \|f(x_n)\| \cdot \|f(x_{n-1})\|}{\beta_n \|f(x_{n+1})\|^2} \right), \quad (9)$$

$$\gamma_{n+1} = \frac{\gamma_n \|f(x_n)\| \cdot \|f(x_{n-1})\|}{\|f(x_{n+1})\| \cdot \|f(x_{n+2})\|} \frac{\beta_{n+1}}{\beta_n}, \quad \gamma_0 = \beta_0^2 \frac{\|f(x_0)\|}{\|f(x_1)\|},$$

$$\beta_{n+1} = \min \left(1, \frac{\gamma_n \|f(x_n)\| \cdot \|f(x_{n-1})\|}{\beta_n \|f(x_{n+1})\|^2} \right), \quad \gamma_{n+1} = \frac{\gamma_n \|f(x_{n-1})\|}{\|f(x_{n+2})\|} \frac{\beta_{n+1}}{\beta_n}, \quad \gamma_0 = \beta_0^2 \frac{\|f(x_0)\|}{\|f(x_1)\|}. \quad (10)$$

$$\beta_{n+1} = \min \left(1, \frac{\gamma_n \|f(x_n)\| \cdot \|f(x_{n-1})\|}{\beta_n \|f(x_{n+1})\|^2} \right), \quad \gamma_{n+1} = \frac{\gamma_n \|f(x_{n-1})\| \cdot \|f(x_{n+1})\|}{\|f(x_{n+2})\|^2} \frac{\beta_{n+1}}{\beta_n}, \quad (11)$$

$$\gamma_0 = \beta_0^2 \frac{\|f(x_0)\|^2}{\|f(x_1)\|^2}.$$

одношаговый метод Жанлава-Пузынина[1], где β_n просчитывается по формуле (12):

$$\beta_{n+1} = \min\left(1, \frac{\beta_n \|f(x_n)\|}{\|f(x_{n+1})\|}\right), \quad (12)$$

и двухшаговый метод, рассмотренный в работе [2], где β_n просчитывается по формуле (13):

$$\beta_{n+1} = \min\left(1, \frac{\gamma_n \|f(x_n)\|}{\beta_n \|f(x_{n+1})\|}\right), \quad \gamma_{n+1} = \frac{\gamma_n \|f(x_n)\|}{\|f(x_{n+2})\|}, \quad \gamma_0 = \beta_0^2. \quad (13)$$

Причем $\|f(x_{-1})\|$ полагаем равным $\|f(x_0)\|$.

Таким образом, нами рассматриваются процессы:

(2)-(4),(9); (2),(3),(5),(10); (2),(3),(7),(11); (2),(3),(6),(12); (2),(3),(6),(13).

Вычислительный эксперимент и его обсуждение.

Вычислительный эксперимент на модельной системе (2) проводился при следующих начальных данных: точность $\varepsilon = 10^{-12}$, $\beta_0 = 10^{-2}$, начальные приближения берутся с отрезка $[-3,3]$. Размерности системы = 50,70,100,150.

Вычислительный эксперимент №1 состоял в следующем: запускались рассмотренные выше методы при одних и тех же начальных условиях, описанных выше, и эксперимент показал следующее: из рассмотренных одношаговых, двухшаговых и трехшаговых методов наиболее эффективными по скорости сходимости к решению является трехшаговый метод (2)-(4),(9) и одношаговый метод (2),(3),(6),(12). А по широте области сходимости наиболее эффективными являются (2),(3),(6),(12), далее идет метод (2)-(4),(9).

Результаты эксперимента представлены в таблице 1.

Таблица 1 – Результаты вычислительного эксперимента

n - Размерность	50	70	100
Метод №1	44.92%	41.52%	36.76%
Метод №2	45.8%	38.8%	25.52%
Метод №3	43.24%	36.44%	35.48%
Метод №4	45.96%	42.36%	38.56%
Метод №5	43.42%	37.27%	31.46%

В вычислительный эксперименте №2 мы проводили вычисления всех методов при одном и том же начальном приближении, а β_0 находилось следующим образом: вначале брали β_0 случайным образом из отрезка $[10^{-3}, 10^{-1}]$, затем, после нахождения Δx_0 , вычисляли x_1 в соответствии с одной из формул (4)-(7), то-есть

$$x_1 = x_0 + \beta_0 \Delta x_0.$$

Далее находили норму, соответствующую данному значению x_1 :

$$\|f(x_n)\| = \sqrt{\sum_{i=1}^n f_i^2(x_n)}.$$

После этого вместо β_0 брали $\frac{\beta_0}{2}$, затем заново вычисляли x_1 и сравнивали его норму с нормой $f(x_1)$, где x_1 найдено на предыдущем шаге.

Если норма $f(x_1)$ меньше, то мы брали значение x_1 за эталон и повторяли все шаги с уменьшением β_0 и сравнением норм заново. А в противном случае в качестве начального параметра принимается β_0 .

Всего таких шагов проводилось 10.

Результаты эксперимента показали незначительное увеличение процента сходимости всех методов.

Вычислительном эксперимент №3 состоял в следующем: для каждого метода мы сохранили значения векторов x_0 , для которых данный метод не сошелся. Далее мы повторяли вычисления, принимая за начальные приближения вектора равные разности двух «плохих» векторов. Результаты эксперимента показали, что среди векторов начальных приближений, найденных таким образом, сходимость стабильно более (примерно на 10%) высокая по сравнению с векторами, взятыми случайным образом.

В вычислительном эксперименте №4 мы сохраняли «плохие» значения вектора x_0 наиболее эффективного из методов – трехшагового метода (2)-(4),(9) – и использовали данные значения как начальные приближения для одного из наименее эффективных методов – метода (2),(3),(7),(11). Результат вычислительного эксперимента показал, что в ряде случаев эти «плохие» вектора являются вполне хорошим начальным приближением для другого метода. Таким образом, комбинированное использование двух различных методов позволяет расширить область сходимости. Это говорит о том, что области сходимости различных методов есть множества пересекающиеся.

Литература.

1. Жанлав, Т. О сходимости на основе непрерывного аналога метода Ньютона / Т. Жанлав, И.В. Пузынин // Вычисл. матем. и матем. физ. – 1992. –Т.32, №6. – С.846-856.
2. Мадорский, В.М. Квазиньютоновские процессы решения нелинейных уравнений: монография / В.М. Мадорский. – Брест: Изд-во БрГУ, 2005. – 186 с.
3. Артющеня, А.А. Сравнительный анализ эффективности одношаговых, двухшаговых и трёхшаговых методов решения нелинейных систем / А. А. Артющеня // Материалы III международной молодежной научно-практической конференции «Научный потенциал молодежи – будущему Беларуси» Пинск : Изд-во Полесского ГУ. – 2009. – Ч. 3. – С. 210.

УДК 517.925

О ПОСТРОЕНИИ ОБЩЕГО ИНТЕГРАЛА У НЕЛИНЕЙНЫХ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ 4-ГО ПОРЯДКА

Белемук О.В.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

В работах [1-3] рассматривается метод построения общего интеграла специальной формы для нелинейных обыкновенных дифференциальных уравнений второго и третьего порядков. Приведенные в этих работах алгоритмы поиска коэффициентных условий, при выполнении которых рассматриваемое дифференциальное уравнение имеет заданный общий интеграл, распространены на уравнения 4-го порядка и реализованы в кодах системы символьных вычислений *Mathematica* 6.03.

Сущность рассматриваемого метода состоит в следующем: рассмотрим функцию

$$\varphi_5(x) = \sum_{i=1}^4 C_i \varphi_i(x) e^{\lambda_i y(x)}, \quad (1)$$

где C_i ($i = \overline{1,4}$) – произвольные постоянные, φ_j ($j = \overline{1,5}$) – произвольные, отличные от нуля аналитические функции, λ_i ($i = \overline{1,4}$) – некоторые постоянные. Сформулируем следующую задачу: найти дифференциальное уравнение 4-го порядка, общий интеграл которого имеет вид (1).

Введем в рассмотрение функции ξ_i ($i = \overline{1,4}$) :

$$\xi_i = \frac{\varphi_i'}{\varphi_i} - \frac{\varphi_5'}{\varphi_5}, \quad (2)$$

в этом случае функции φ_i ($i = \overline{1,4}$) примут вид:

$$\varphi_i = \varphi_5 e^{\int \xi_i dx}. \quad (3)$$

После подстановки (3) в функцию (1) получим равенство:

$$\sum_{i=1}^4 C_i e^{\lambda_i y(x) + \int \xi_i dx} = 1, \quad (4)$$

которое будет представлять общий интеграл дифференциального уравнения 4-го порядка.

Алгоритм построения дифференциального уравнения, для которого соотношение (4) будет общим интегралом, проводится в два этапа: 1) построение дифференциального уравнения с неопределенными коэффициентами; 2) установление соотношений, связывающих коэффициенты построенного дифференциального уравнения.

Результатом первого этапа построения алгоритма является дифференциальное уравнение 4-го порядка

$$\begin{aligned} & \left(\sum_{i=0}^2 (a_{i101} y'^i) y'' + \sum_{i=0}^5 (a_{i001} y'^i) \right) y^{(IV)} + \left(\sum_{i=0}^4 (a_{i110} y'^i) y'' + \sum_{i=0}^6 (a_{i010} y'^i) \right) y''' + \\ & + a_{0210} y''^2 y''' + \sum_{i=0}^2 (a_{i020} y'^i) y'''^2 + a_{0400} y''^4 + \sum_{i=0}^3 (a_{i300} y'^i) y'''^3 + \\ & \sum_{i=0}^5 (a_{i200} y'^i) y''^2 + \sum_{i=0}^7 (a_{i100} y'^i) y'' + \sum_{i=0}^{10} (a_{i000} y'^i) = 0, \end{aligned} \quad (5)$$

где a_{ijkl} - коэффициенты при $y'^i y''^j y'''^k y^{(IV)l}$, $i, j, k, l \in \mathbb{Z}$, $i = \overline{0,10}$, $j = \overline{0,4}$, $k = \overline{0,2}$, $l = 0,1$.

Реализованный средствами ССВ *Mathematica* алгоритм позволил установить вид коэффициентов дифференциального уравнения (5), линейные зависимости между коэффициентами при одинаковых весовых характеристиках, а также вид функций ξ_i ($i = \overline{1,4}$), что позволяет классифицировать дифференциальные уравнения вида (5) в зависимости от вводимых коэффициентов и строить для них общий интеграл.

Литература

1. Belemuk O., Chichurin A. Cauchy Problem for Abel's Differential Equation of the First Kind // Computer Algebra Systems in Teaching and Research. Evolution, Control and Stability of Dynamical Systems. WSFiZ, Siedlce, 2009. P. 18-22.

2. Белемук, О.В. О построении общего интеграла у обыкновенных дифференциальных уравнений второго порядка специального вида с помощью компьютерного моделирования / О.В. Белемук, А.В. Чичурин // Book of abstracts of the Second International Conference for Young Mathematicians on Differential Equations and Applications dedicated to Ya. B. Lopatinskii (DNU, Donetsk, Ukraine, 11-14 November 2008). – Donetsk, Цифрова типографія, 2008. – С. 42-43.

3. Белемук, О.В. Алгоритмический и программный аспекты решения нелинейных обыкновенных дифференциальных уравнений специального вида / О.В. Белемук // Современные проблемы математического моделирования и новые образовательные технологии в математике: материалы республиканской научно-практической конференции молодых ученых, аспирантов и студентов, БрГУ, 22-23 апреля 2009 г. – Том 1. – Брест: Альтернатива. – 2009. – С. 68-70.

УДК 517.983+519.6

СХОДИМОСТЬ ОДНОГО МЕТОДА РЕШЕНИЯ ЛИНЕЙНЫХ УРАВНЕНИЙ С НЕОГРАНИЧЕННЫМ ОПЕРАТОРОМ В ГИЛЬБЕРТОВОМ ПРОСТРАНСТВЕ

Берёзкина М.С.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Савчук В.Ф., кандидат физ.-мат. наук, доцент*

Поскольку некорректные задачи постоянно возникают в многочисленных приложениях математики, то проблема их решения и разработка методов их решения является актуальной. В работе предлагается новый метод решения некорректных задач. Цель исследования – доказать сходимость метода и получить оценки погрешности в исходной норме гильбертова пространства.

В гильбертовом пространстве H решается уравнение I рода

$$Ax = y, \quad (1)$$

где A – неограниченный положительный самосопряжённый оператор, для которого нуль не является собственным значением. Однако нуль принадлежит спектру оператора A и, следовательно, задача о разрешимости уравнения (1) является некорректной. Если решение уравнения (1) существует, то будем искать его с помощью итерационного метода

$$x_{n+1} = (A^2 + B)^{-1}(Bx_n + Ay), \quad x_0 = 0. \quad (2)$$

Здесь B – ограниченный вспомогательный самосопряжённый оператор, который выбирается для улучшения обусловленности. В качестве B возьмём оператор $B = bE$, $b > 0$, E – тождественный оператор. В случае приближённой правой части y_δ , $\|y - y_\delta\| \leq \delta$, итерационный процесс (2) запишется в виде

$$x_{n+1} = (A^2 + B)^{-1}(Bx_{n,\delta} + Ay_\delta), \quad x_{0,\delta} = 0. \quad (3)$$

Для метода (3) доказаны теоремы.

Теорема 1. Итерационный процесс (2) сходится к точному решению x уравнения (1).

Теорема 2. Если выбирать число итераций n в зависимости от δ так, чтобы $\sqrt{n}\delta \rightarrow 0$, $n \rightarrow \infty$, $\delta \rightarrow 0$, то итерационный процесс (3) сходится.

Теорема 3. Если точное решение x уравнения (1) истокорпредставимо, т.е. $x = A^{2s} z$, $s > 0$, то для итерационного процесса (3) имеет место оценка погрешности:

$$\|x - x_{n,\delta}\| \leq \left(\frac{bs}{2n}\right)^s \|z\| + \sqrt{\frac{n}{2b}} \delta, \quad n > 2s. \quad (4)$$

Теорема 4. Оптимальная по n оценка погрешности для метода (3) имеет вид

$$\|x - x_{n,\delta}\|_{opt} \leq \left[(1 + 2s) 2^{-4s} s^{-s} \|z\|^2 \delta^{2s} \right]^{1/(2s+1)}. \quad (5)$$

и получается при $n_{opt} = \left[2^{3-2s} s^{2(s+1)} b^{2s+1} \|z\|^2 \delta^{-2} \right]^{1/(2s+1)}$.

Замечание. Если оператор A – ограниченный, $A = \int_{-M}^M \lambda dE_\lambda$, то следует учитывать значение функции $f(\lambda)$ в точке $\lambda = M$, поэтому общая оценка погрешности метода (3) будет иметь вид

$$\|x - x_n\| \leq \max \left\{ \left(\frac{bs}{2n}\right)^s, \frac{M^{2s} b^n}{(M^2 + b)^n} \right\} \|z\| + \sqrt{\frac{n}{2b}} \delta, \quad n > 2s. \quad (6)$$

Очевидно, при $n \rightarrow \infty$ величина $\frac{M^{2s} b^n}{(M^2 + b)^n}$, убывающая как геометрическая прогрессия, станет меньше величины $\left(\frac{bs}{2n}\right)^s$, убывающей как $\frac{1}{n^s}$. Следовательно, для достаточно больших n оценка (6) примет вид (4).

Предложенный метод может быть успешно применён для решения некорректных задач, встречающихся в спектроскопии, акустике, синтезе антенн, математической обработке данных эксперимента, теплопроводности.

УДК 517.512.2

АСИМПТОТИЧЕСКИЕ ФОРМУЛЫ ТИПА ВОРОНОВСКОЙ

Божко И.Н., Дацьк В.Т.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

Одной из основных задач теории суммирования рядов и интегралов, является нахождение главного члена уклонения функций определенного класса от ее линейных средних (операторов приближения) с равномерной оценкой остатка относительно всего класса указанных функций. Впервые такая задача была решена Е.В. Вороновской, а именно:

$$f(x) - B_n(f; x) = -\frac{1}{2} \frac{x(1-x)}{n} f(x) + O\left(\frac{1}{n}\right) \quad (1)$$

Теорема 1. Если функция $f(x)$ абсолютно интегрируема и ограничена на числовой прямой, то справедлива следующая асимптотическая формула

$$F_\sigma(f; x) - f(x) = \frac{1}{\pi} \int_{-\lambda}^{\infty} \frac{f\left(x - \frac{t}{\sigma}\right) - 2f(x) + f\left(x + \frac{t}{\sigma}\right)}{t^2} dt + O(\omega_2(\sigma^{-1}; f)) \quad (2)$$

где

$$F_\sigma(f; x) = \frac{1}{\pi} \int_0^\sigma \left(1 - \frac{u}{\sigma}\right) du \int_{-\infty}^{\infty} f(t) \cos u(x-t) dt \quad (3)$$

есть $(C,1)$ -средние (средние Фейера) интеграла Фурье функции $f(t)$;

$\omega_2(\sigma^{-1}; f) = \sup_{|t| \leq \sigma^{-1}} \max_{x \in R} |f(x-t) - 2f(x) + f(x+t)|$ - модули гладкости второго порядка функции f ;

$\lambda > 0$ - произвольная действительная постоянная.

Следствие 1. Если дополнительно к условиям теоремы $f(x)$ есть ещё и функция класса Гёльдера порядка $0 < \alpha \leq 1$, то из определения модуля гладкости $\omega_2(\frac{1}{\sigma}; f)$ следует, что:

$$F_\sigma(f; x) - f(x) = \frac{1}{\pi} \int_{-\lambda}^{\infty} \frac{f\left(x - \frac{t}{\sigma}\right) - 2f(x) + f\left(x + \frac{t}{\sigma}\right)}{t^2} dt + O(\sigma^{-\alpha}), \quad (4)$$

причём, если $0 < \alpha < 1$, то и интеграл правой части (4) также имеет порядок $O(\sigma^{-\alpha})$.

Обозначим через $W^{(2\rho+1)}(D)$ (ρ - фиксированное целое неотрицательное число) класс абсолютно интегрируемых на числовой прямой функций f вместе со своими существующими $(2\rho + 1)$ -первыми производными, причём $|f^{(2\rho+1)}(t)| \leq D < \infty$

Видно, что все производные до порядка 2ρ включительно, а также сама функция $f(t)$, принадлежат классу Липшица порядка $\alpha = 1$. Тогда для функций введенного класса $W^{(2\rho+1)}(D)$ справедливы представления.

$$f^{(m)}(x) = \frac{1}{\pi} \int_0^\infty u^m du \int_{-\infty}^{\infty} f(t) \cos\left(u(x-t) + \frac{m\pi}{2}\right) dt \quad (5)$$

$$\overline{f^{(m)}}(x) = \frac{1}{\pi} \int_0^\infty u^m du \int_{-\infty}^{\infty} f(t) \cos\left(u(x-t) + \frac{m+1}{2}\pi\right) dt \quad (6)$$

$$m = \overline{1, 2\rho}$$

Введём обобщённые средние сопряжённого интеграла Фурье.

$$\overline{U}_\sigma(f; x) := -\frac{1}{\pi} \int_0^\sigma K(\sigma, u) du \int_{-\infty}^{\infty} f(t) \sin u(x-t) dt \quad (7)$$

$$\text{где } K(\sigma, u) := \sum_{m=0}^{\infty} a_m(\sigma) \left(\frac{u}{\sigma}\right)^m \quad (8)$$

есть сумма или абсолютно сходящийся ряд по степеням $\frac{u}{\sigma}$, $0 \leq u \leq \sigma$, $\sigma > 0$. Причём, коэффициенты $a_m(\sigma)$ такие, что ряд

$$A_\sigma := a_0(\sigma) + \sum_{m=1}^{\infty} m|a_m(\sigma)| \quad (9)$$

сходится.

Справедлива теорема:

Теорема 2. Если $f \in W^{(2\rho+1)}(D)$, то

$$\begin{aligned} \bar{U}_\sigma(f; x) = & \sum_{\nu=0}^{\rho} (-1)^\nu \bar{f}^{(2\nu)}(x) \frac{a_{2\nu}(\sigma)}{\sigma^{2\nu}} + \sum_{\nu=1}^{\rho+1} (-1)^{\nu+1} f^{(2\nu-1)}(x) \frac{a_{2\nu-1}(\sigma)}{\sigma^{2\nu-1}} + \\ & + \frac{(-1)^\rho}{\sigma^{2\rho+1}} J_{\sigma, \rho}(x) \sum_{m=0}^{\infty} a_m(\sigma) + \frac{1}{\sigma^{2\rho+1}} (O(\omega_2(\sigma^{-1}; f^{(2\rho+1)})) + B_{\sigma, \rho}) A_\sigma \end{aligned} \quad (10)$$

где $B_{\sigma, \rho} = \mu(1)$;

$$J_{\sigma, \rho}(x) := \frac{1}{\pi} \int_0^\infty \Phi_x^{(2\rho+1)}(t) \frac{\sin \sigma t}{t} dt,$$

$$\Phi_x^{(2\rho+1)}(t) := f^{(2\rho+1)}(x+t) - 2f^{(2\rho+1)}(x) + f^{(2\rho+1)}(x-t)$$

Следствие 2. Если $f \in W^{(2\rho+1)}(D)$ и удовлетворяет условию Гельдера порядка α ,

$$0 < \alpha < 1, \text{ то в формуле (10) будет } e_{\sigma, \rho} = \left(\frac{1}{\sigma^\alpha}\right) \text{ и } \mu(\omega_2(\sigma^{-1}; f^{(2\rho+1)})) = O\left(\frac{1}{\sigma^\alpha}\right).$$

Следствие 3. Для методов суммирования, у которых $\sum_m 0^{\alpha_m}(\sigma) = 0$, например:

средних Зигмунда с $K_{(\sigma, u)} = 1 - \left(\frac{u}{\sigma}\right)^8$, слагаемое с множителем $J_{\sigma, \rho}(x)$ в формуле (10) будет отсутствовать. И в этом случае, например, при выполнении условий следствия 1, будем иметь (если $s > 2\rho + 1$)

$$\bar{U}_\sigma(f; x) = \overline{f(x)} + O\left(\frac{1}{\sigma^{2\rho+1+\alpha}}\right)$$

Литература

- 1 Дзядык, В.К. Введение в теорию равномерного приближения функций полиномами / В.К. Дзядык – М.: Наука, 1977.
- 2 Ибрагимов, И.И. Экстремальные свойства функций конечной степени / И.И. Ибрагимов - Баку, 1962.
- 3 Семенчук Н.П. Асимптотические формулы типа Вороновской / Н.П. Семенчук, В.Т. Дачык // Весці АН БССР. – 1996. – № 2.

НОРМАЛИЗАЦИЯ СИСТЕМ ЛИНЕЙНЫХ ГАМИЛЬТОНОВЫХ УРАВНЕНИЙ

Будько Д.А.

УО «Брестский государственный университет им. А.С. Пушкина», г. Брест

1 Введение. Хорошо известно, что анализ систем нелинейных дифференциальных уравнений начинают с исследования соответствующей линеаризованной системы уравнений. Рассмотрим автономную систему линейных гамильтоновых уравнений:

$$\frac{dx}{dt} = J H x, \quad (1)$$

где $x = (x_1, \dots, x_n, Px_1, \dots, Px_n)$, x_k, Px_k – канонически сопряжённые координаты и импульсы. Матрица H является вещественной, порядка $2n$. Через J обозначена матрица $\begin{pmatrix} 0 & E_n \\ -E_n & 0 \end{pmatrix}$, где E_n единичная матрица порядка n .

Известно [1], что если функция Гамильтона H является знакоопределённой, то на основании теорем Ляпунова об устойчивости [1], вопрос об устойчивости будет решён в строгой нелинейной постановке. Таким образом, не требуется анализировать члены более высоких порядков в нелинейных уравнениях возмущённого движения, а знакоопределённость функции можно определить после приведения её к нормальной форме. Рассмотрим один из алгоритмов нормализации системы, предложенный в [2]. Хотя идея алгоритма достаточно проста, при реализации возникает немало трудностей и нюансов, которым и посвящена данная работа. Приведена реализация алгоритма нормализации системы линейных гамильтоновых уравнений в случае $n = 2$ в кодах системы компьютерной алгебры *Mathematica* [3].

В качестве примера рассмотрим функцию Гамильтона, которая описывает следующую модель космической динамики. Три тела движутся по круговым орбитам вокруг общего центра масс системы, образуя правильный треугольник в любой момент времени. Четвёртое тело, пренебрежимо малой массы, движется в гравитационном поле, генерируемое тремя массивными телами, оставаясь в их орбитальной плоскости. Такая задача известна под названием плоской ограниченной задачи четырёх тел, сформулированной на основе треугольных решений Лагранжа [4] и обладает двумя степенями свободы ($n = 2$). При разложении функции Гамильтона в ряд Тейлора в окрестности равновесного решения первым (линейным) приближением будет квадратичная часть H_2 функции Гамильтона:

$$H_2 = \frac{Px_1^2 + Px_2^2}{2} - Px_2 x_1 + Px_1 x_2 + h_{11} x_1^2 + h_{12} x_1 x_2 + h_{22} x_2^2 \quad (2)$$

где коэффициенты h_{11}, h_{12}, h_{22} являются константами при фиксированных значениях параметров модели. Функции (2) соответствует матрица H линейной системы уравнений (1):

$$H = \begin{pmatrix} 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 \\ -2h_{11} & -h_{12} & 0 & 1 \\ -h_{12} & -2h_{22} & -1 & 0 \end{pmatrix} \quad (3)$$

2 Приведение линейной системы к нормальной форме. Предположим, что характеристические показатели системы являются чисто мнимыми числами, и для матрицы (3) они равны

$$\lambda = \pm i \sigma_{1,2}, \quad \sigma_{1,2} = \sqrt{1 + h_{11} + h_{22} \pm \sqrt{4h_{11} + h_{11}^2 + h_{12}^2 + 4h_{22} - 2h_{11} h_{22} + h_{22}^2}} \quad (4)$$

Как известно, нормальной формой системы уравнений (1) является такая система, которой соответствует функция Гамильтона, равная алгебраической сумме n линейных, не связанных между собой осцилляторов. Следовательно, нормальная форма системы уравнений (1) запишется в виде:

$$\frac{dy}{dt} = J \hat{H} y, \quad (5)$$

где \hat{H} – вещественная, диагональная матрица, элементы которой $\hat{h}_{kk} = \hat{h}_{k+n,k+n} = \sigma_k$.
Переход от старых переменных x к новым y определим с помощью матрицы A в виде равенства

$$x = A y, \quad y = (p_1, \dots, p_n, q_1, \dots, q_n) \quad (6)$$

Согласно предложенному алгоритму, k -столбцом матрицы преобразования A являются вектор $-2s_k$, а $(k+n)$ -столбцом – вектор $2r_k$. При этом r_k и s_k являются вещественной и мнимой частями k -го собственного вектора матрицы JH . Так как собственные вектора определяются с точностью до константы c_k , то в алгоритме предусмотрено условие выбора этих констант, при котором преобразование сохранялось бы вещественным:

$$4(r_k, J s_k) = 1, \quad k = 1, 2, \dots, n. \quad (7)$$

Приведём программу для нахождения матрицы преобразования A при $n = 2$, написанную в кодах системы *Mathematica*:

```
rulσ = {σ1, σ2} → (Eigenvalues[J.H][[2,4]])/i//Thread//Simplify};
X = {x1, x2, x3, x4};
eqEigen = J.H.X == λ X//Thread;
solEigen = Solve[eqEigen[[{1,2,3}],X[[{2,3,4}]]][[1]]//Simplify;
vectors = ck X /.solEigen/.x1 → 1/λ → i σk;
st := ComplexExpand[ Im[vectors/.k → t]]//Simplify;
rt := ComplexExpand[ Re[vectors/.k → t]]//Simplify;
rulck := {ck → √Solve[(4rk.(J.sk) == 1)/.ck2 → t,t][[1,1,2]]} /.σ2 → -σ2//
Simplify//Flatten;
A = Transpose[{ -2s1, -2s2, 2r1, 2r2 } /.rulc1 /.rulc2 /.σ2 → -σ2 /.rulσ
```

С помощью встроенной функции *Eigenvalues* находим собственные значения матрицы JH , выбираем σ_k положительными. Далее находим собственные вектора матрицы JH из уравнения *eqEigen*. Выделяем вещественную и мнимую части собственного вектора в переменные r_t и s_t . Потом записываем в правило *rulc_k* найденные константы c_k из условия нормировки (7) и в итоге записываем матрицу нормализующего преобразования A .

Часто при анализе гамильтоновых систем оказывается, что при выборе всех $\sigma_k > 0$ хотя бы одно из чисел c_k^2 является отрицательным, что противоречит вещественности преобразования (6). Однако можно заметить, что знак c_k^2 прямо пропорционально зависит от знака σ_k , поэтому если изменить собственное значение $i\sigma_k$ на $-i\sigma_k$, то c_k^2 будет положителен. Естественно, надо выбирать и собственный вектор, соответствующий собственному числу $-i\sigma_k$.

В нашем случае следует выбирать собственное значение $-i\sigma_2$ и соответствующий собственный вектор, тогда c_2^2 будет положительным. Поэтому в правиле *rule_k* и матрице A мы заменяем $\sigma_2 \rightarrow -\sigma_2$. В результате получаем матрицу нормализующего преобразования переменных A , которая является невырожденной, вещественной и симплектической.

Функция Гамильтона, соответствующая нормальной форме линейной системы уравнений (1), при $n = 2$ принимает вид

$$H_2 = \frac{1}{2}(\sigma_1(p_1^2 + q_1^2) - \sigma_2(p_2^2 + q_2^2)). \quad (8)$$

Матрица A для функции Гамильтона (2) принимает вид:

$$A = \begin{pmatrix} 0 & 0 & 2c_1 & 2c_2 \\ -2\sigma_1 u_1 & 2\sigma_2 u_2 & h_{12} u_1 & h_{12} u_2 \\ -f_1 & f_2 & -h_{12} u_1 & -h_{12} u_2 \\ -h_{12} \sigma_1 u_1 & h_{12} \sigma_2 u_2 & g_1 & g_2 \end{pmatrix}. \quad (9)$$

Коэффициенты u_k, f_k, g_k, d_k в матрице (9) определены формулами:

$$u_k = \frac{2c_k(1 - 2h_{11} + \sigma_k^2)}{d_k}, \quad f_k = \frac{2c_k \sigma_k(-2 + 4h_{11} + h_{12}^2 + 2\sigma_k^2)}{d_k},$$

$$g_k = \frac{2c_k(h_{12}^2 + (2 + 4h_{11})\sigma_k^2 - 2\sigma_k^4)}{d_k}, \quad d_k = h_{12}^2 + 4\sigma_k^2.$$

Заключение. В данной работе приведена практическая реализация алгоритма для нахождения нормальной формы системы линейных гамильтоновых уравнений. В качестве примера рассмотрена нормализация квадратичной части функции Гамильтона (2). Данной функцией Гамильтона описывается плоская ограниченная задача четырёх тел, сформулированная на основе треугольных решений Лагранжа. Построена матрица преобразования (9) для функции (2), которая приводит её к нормальной форме (8).

Легко видеть, что (8) не является знакоопределенной функцией, поэтому вопрос об устойчивости по Ляпунову не может быть решён в первом приближении. Поэтому в данном случае для исследования устойчивости решений в строгой нелинейной постановке необходимо анализировать члены более высоких порядков в разложении функции Гамильтона.

Все символьные и численные вычисления выполнены с помощью системы компьютерной алгебры *Mathematica*.

Литература

1. Ляпунов, А.М. Общая задача об устойчивости движения / Под ред. Мюнтц Г. – Череповец: Меркурий-Пресс, 2000. – 386 с.
2. Маркеев, А.П. Точки либрации небесной механике и космодинамике / А. П. Маркеев – М.: Наука, 1978. – 312 с.
3. Прокопеня, А.Н. Применение системы Mathematica к решению обыкновенных дифференциальных уравнений / А.Н. Прокопеня, А.В. Чичурин. – Минск: БГУ, 1999. – 265 с.
4. Budzko, D.A. Linear stability analysis of equilibrium solutions of restricted planar four-body problem/ D.A. Budzko // Computer Algebra Systems in Teaching and Research: proc. of the 5th International Workshop CASTR'2009, Siedlce, Poland, 28 – 31 Jan., 2009 / University of Podlasie; Eds.: L. Gadomski [and others]. – Siedlce, 2009. – P. 28–36.

УДК 517

ПРЕДСТАВЛЕНИЕ НЕКОТОРЫХ ФУНКЦИЙ ДВУХ ПЕРЕМЕННЫХ СИНГУЛЯРНЫМИ ИНТЕГРАЛАМИ В ТОЧКАХ ПЕРЕСЕЧЕНИЯ ИХ ЛИНИЙ РАЗРЫВА

Велесевич А.И., Дацьк В.Т.

УО «Брестский государственный технический университет», г. Брест

В работе рассматриваются сингулярные интегралы

$$f_{m,n}^{(x,y)} = \int_b^a \int_c^d f(t, v) \Phi_m(t-x) \Phi_n(v-y) dt dv, \tag{1}$$

у которых ядро $\Phi_m(t-x)\Phi_n(v-y)$ ограничено. Тогда двойной интеграл существует для любой суммируемой функции двух переменных.

Выделяются классы функций, для которых вычисляется $\lim_{m,n \rightarrow \infty} f_{mn}(x, y)$, где (x,y) — точка пересечения линий разрыва функции $f(t, v)$.

Теорема. Пусть (x,y) — точка пересечения линий, разрыва функций $f(t, v)$, определяемых монотонными гладкими функциями $v = \mu(t)$ и $y = \psi(t)$, причем:

$$\begin{aligned} a &\leq t \leq b \\ a &< x < b \\ c &\leq v \leq d \\ c &< y < L \end{aligned} \tag{2}$$

Если выполнимы условия:

$$\lim_{u,z \rightarrow 0} \frac{f[L(u, z); b(u, z)] + f[L(u, -z); b(u, -z)] + f[L(-u, -z); b(-u, -z)] + f[L(-u, z); b(-u, z)]}{4} = A, \tag{3}$$

где $A \in (-\infty, +\infty)$.

При любых $\delta_1 > 0$ и $\delta_2 > 0$ ядро $\Phi_m(t-x)\Phi_n(v-y)$ слабо сходится к нулю в области $\Gamma = [a,b;c,d] - \Delta$,

где Δ — криволинейный параллелограмм NBCD с координатами вершин

$$N [L(\delta_1, \delta_2), \beta(\delta_1, \delta_2)]; B [L(-\delta_1, \delta_2), \beta(-\delta_1, \delta_2)];$$

$$C [L(-\delta_1, -\delta_2), \beta(-\delta_1, -\delta_2)]; D [L(\delta_1, -\delta_2), \beta(\delta_1, -\delta_2)].$$

$$\int_a^b \int_c^d |\Phi_m(t-x)\Phi_n(v-y)dt dv| < C < \infty \quad (4)$$

где C не зависит от m и n .

Существуют $\delta'_0 > 0$ и $\delta''_0 > 0$, что:

$$\begin{aligned} \text{а)} & |J_1|\Phi_m[L(u, z) - x]\Phi_n[\beta(u, z) - y] + |J_2|\Phi_m[L(u, -z) - x]\Phi_n[\beta(u, -z) - y] - \\ & - |J_3|\Phi_m[L(-u - z) - x]\Phi_n[\beta(-u, -z) - y] - |J_4|\Phi_m[L(-u, z) - x]\Phi_n[\beta(-u, z) - y]| < M_1 \\ \text{б)} & |J_1|\Phi_m[L(u, z) - x]\Phi_n[\beta(u, z) - y] - |J_2|\Phi_m[L(u, -z) - x]\Phi_n[\beta(u, -z) - y] + \\ & + |J_3|\Phi_m[L(-u - z) - x]\Phi_n[\beta(-u, -z) - y] - |J_4|\Phi_m[L(-u, z) - x]\Phi_n[\beta(-u, z) - y]| < M_2 \\ \text{в)} & |J_1|\Phi_m[L(u, z) - x]\Phi_n[\beta(u, z) - y] - |J_2|\Phi_m[L(u, -z) - x]\Phi_n[\beta(u, -z) - y] - \\ & - |J_3|\Phi_m[L(-u - z) - x]\Phi_n[\beta(-u, -z) - y] + |J_4|\Phi_m[L(-u, z) - x]\Phi_n[\beta(-u, z) - y]| < M_3 \\ \text{г)} & -|J_1|\Phi_m[L(u, z) - x]\Phi_n[\beta(u, z) - y] + |J_2|\Phi_m[L(u, -z) - x]\Phi_n[\beta(u, -z) - y] + \\ & + |J_3|\Phi_m[L(-u - z) - x]\Phi_n[\beta(-u, -z) - y] - |J_4|\Phi_m[L(-u, z) - x]\Phi_n[\beta(-u, z) - y]| < M_4 \\ \text{д)} & -|J_1|\Phi_m[L(u, z) - x]\Phi_n[\beta(u, z) - y] + |J_2|\Phi_m[L(u, -z) - x]\Phi_n[\beta(u, -z) - y] - \\ & - |J_3|\Phi_m[L(-u - z) - x]\Phi_n[\beta(-u, -z) - y] + |J_4|\Phi_m[L(-u, z) - x]\Phi_n[\beta(-u, z) - y]| < M_5 \\ \text{ж)} & -|J_1|\Phi_m[L(u, z) - x]\Phi_n[\beta(u, z) - y] - |J_2|\Phi_m[L(u, -z) - x]\Phi_n[\beta(u, -z) - y] + \\ & + |J_3|\Phi_m[L(-u - z) - x]\Phi_n[\beta(-u, -z) - y] + |J_4|\Phi_m[L(-u, z) - x]\Phi_n[\beta(-u, z) - y]| < M_6 \end{aligned} \quad (5)$$

где M_k ($k = 1, 2, \dots, 6$) не зависит от m и n , а $u < \delta'_0, z < \delta''_0$, то

$$\lim_{m, n \rightarrow \infty} f_{mn}(x, y) = A \quad (6)$$

Замечания:

1. Теорема легко переносится на случай, если через рассматриваемую точку (x, y) проходит n линий (n — любое конечное не отрицательное целое число). Если $n = 0$, то точка (x, y) может быть точкой разрыва или точкой непрерывности функции $f(t, v)$

2. Теорема применяется к широкому классу линейных операторов, например классических сингулярных интегралов Пуассона, Джексона, Ландау, Валле-Пуссена и др.

Литература

1. Коровкин П.П. Линейные операторы и теория приближений / П.П. Коровкин — М., 1959.

УДК 551.492

СХОДИМОСТЬ И СВОЙСТВА РЕШЕНИЯ ОДНОГО СТОХАСТИЧЕСКОГО УРАВНЕНИЯ

Гетман В.А., Гладкий И.И., Ефимик С.В., Махнист Л.П.

УО «Брестский государственный технический университет», г.Брест

Рассмотрим дифференциальное уравнение для описания колебаний речного стока, используемое в стохастической гидрологии [1]:

$$\frac{d^2\theta_1}{d\xi^2} - \xi \frac{d\theta_1}{d\xi} = -1, \quad \frac{d\theta_1}{d\xi}(\infty) = 0, \quad \theta_1(\xi)|_{\xi=\xi_0} = 0 \quad (1)$$

Уравнение (1) при решении различных прикладных задач, например в [2], интегрировалось численными методами. Приведем решение этого уравнения [3].

Введем обозначение $\frac{d\theta_1}{d\xi} = f(\xi)$. Тогда, учитывая, что $\frac{d^2\theta_1}{d\xi^2} = \frac{df}{d\xi}$, приходим к

линейному дифференциальному уравнению первого порядка $\frac{df}{d\xi} - \xi f = -1$, с начальным условием $f(\xi)|_{\xi=\infty} = 0$.

Решение последнего уравнения будем отыскивать в виде $f(\xi) = u(\xi)v(\xi)$. Тогда, учитывая, что $f'(\xi) = u'(\xi)v(\xi) + u(\xi)v'(\xi)$, получим уравнение $u'v + uv' - \xi uv = -1$ или $u'v + u(v' - \xi v) = -1$ (*).

Найдем одно из ненулевых решений уравнения $v' - \xi v = 0$. Разделяя переменные в уравнении $\frac{dv}{d\xi} = \xi v$, решением которого, очевидно, является $v = 0$, получим $\frac{dv}{v} = \xi d\xi$.

Интегрируя последнее уравнение, имеем $\int \frac{dv}{v} = \int \xi d\xi + C_2$. Откуда $\ln|v| = \frac{\xi^2}{2} + \ln C_1$ или $v = \pm C_1 e^{\frac{\xi^2}{2}}$. Следовательно, $v = C e^{\frac{\xi^2}{2}}$ – общее решение дифференциального уравнения $v' - \xi v = 0$.

Выберем одно из ненулевых решений этого уравнения, например, $v = e^{\frac{\xi^2}{2}}$, при $C = 1$. Подставляя его в уравнение (*), имеем $u'e^{\frac{\xi^2}{2}} = -1$ или $u' = -e^{-\frac{\xi^2}{2}}$. Откуда $u = -\int e^{-\frac{\xi^2}{2}} d\xi + C$.

Следовательно, $f(\xi) = u(\xi)v(\xi) = (-\int e^{-\frac{\xi^2}{2}} d\xi + C)e^{\frac{\xi^2}{2}}$ или

$f(\xi) = (C - \int_{-\infty}^{\xi} e^{-\frac{t^2}{2}} dt)e^{\frac{\xi^2}{2}}$ – общее решение дифференциального уравнения

$$\frac{df}{d\xi} - \xi f = -1.$$

Заметим, что $\sqrt{\frac{2}{\pi}} \int_0^{+\infty} e^{-\frac{t^2}{2}} dt = 1$. Тогда $\int_{-\infty}^{+\infty} e^{-\frac{t^2}{2}} dt = 2 \int_0^{+\infty} e^{-\frac{t^2}{2}} dt = \sqrt{2\pi}$ и, учитывая

начальное условие $f(\xi)|_{\xi=-\infty} = 0$, имеем $f(\xi) = (\sqrt{2\pi} - \int_{-\infty}^{\xi} e^{-\frac{t^2}{2}} dt) e^{\frac{\xi^2}{2}}$.

Действительно, $f(\xi)|_{\xi=-\infty} = \lim_{\xi \rightarrow +\infty} (\sqrt{2\pi} - \int_{-\infty}^{\xi} e^{-\frac{t^2}{2}} dt) e^{\frac{\xi^2}{2}} = \lim_{\xi \rightarrow +\infty} \frac{\sqrt{2\pi} - \int_{-\infty}^{\xi} e^{-\frac{t^2}{2}} dt}{e^{-\frac{\xi^2}{2}}}$

и, учитывая $\int_{-\infty}^{+\infty} e^{-\frac{t^2}{2}} dt = \sqrt{2\pi}$, получаем, что последний предел имеет неопределенность вида $\frac{0}{0}$. Тогда, используя правило Лопиталья, имеем

$$f(\xi)|_{\xi=-\infty} = \lim_{\xi \rightarrow +\infty} \frac{\left(\sqrt{2\pi} - \int_{-\infty}^{\xi} e^{-\frac{t^2}{2}} dt \right)'}{\left(e^{-\frac{\xi^2}{2}} \right)'} = \lim_{\xi \rightarrow +\infty} \frac{0 - e^{-\frac{\xi^2}{2}}}{e^{-\frac{\xi^2}{2}} \cdot (-\xi)} = \lim_{\xi \rightarrow +\infty} \frac{1}{\xi} = 0.$$

Таким образом $f(\xi) = (\sqrt{2\pi} - \int_{-\infty}^{\xi} e^{-\frac{t^2}{2}} dt) e^{\frac{\xi^2}{2}}$ или, учитывая $\int_{-\infty}^0 e^{-\frac{t^2}{2}} dt = \sqrt{\frac{\pi}{2}}$,

$$f(\xi) = \left(\sqrt{2\pi} - \int_{-\infty}^0 e^{-\frac{t^2}{2}} dt - \int_0^{\xi} e^{-\frac{t^2}{2}} dt \right) e^{\frac{\xi^2}{2}} = \left(\sqrt{2\pi} - \sqrt{\frac{\pi}{2}} - \int_0^{\xi} e^{-\frac{t^2}{2}} dt \right) e^{\frac{\xi^2}{2}} =$$

$$= \left(\sqrt{\frac{\pi}{2}} - \int_0^{\xi} e^{-\frac{t^2}{2}} dt \right) e^{\frac{\xi^2}{2}} \text{ — решение дифференциального уравнения } \frac{df}{d\xi} - \xi f = -1,$$

удовлетворяющее начальному условию $f(\xi)|_{\xi=-\infty} = 0$.

Используя разложение функции e^z в ряд Тейлора $e^z = 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \dots = \sum_{n=0}^{+\infty} \frac{z^n}{n!}$,

имеем

$$e^{\frac{\xi^2}{2}} = 1 + \frac{\xi^2}{2} + \frac{\xi^4}{2^2 2!} + \frac{\xi^6}{2^3 3!} + \dots = \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} \text{ и}$$

$$e^{-\frac{t^2}{2}} = 1 - \frac{t^2}{2} + \frac{t^4}{2^2 2!} - \frac{t^6}{2^3 3!} + \dots = \sum_{n=0}^{+\infty} \frac{(-1)^n t^{2n}}{2^n n!}.$$

$$\text{Тогда } \int_0^{\xi} e^{-\frac{t^2}{2}} dt = \int_0^{\xi} \left(\sum_{n=0}^{+\infty} \frac{(-1)^n t^{2n}}{2^n n!} \right) dt = \sum_{n=0}^{+\infty} \frac{(-1)^n t^{2n+1}}{2^n n! (2n+1)} \Big|_0^{\xi} = \sum_{n=0}^{+\infty} \frac{(-1)^n \xi^{2n+1}}{2^n n! (2n+1)}.$$

Следовательно, решение $f(\xi) = \left(\sqrt{\frac{\pi}{2}} - \int_0^\xi e^{-\frac{t^2}{2}} dt \right) e^{\frac{\xi^2}{2}}$ можно представить в виде

$$f(\xi) = \left(\sqrt{\frac{\pi}{2}} - \sum_{n=0}^{+\infty} \frac{(-1)^n \xi^{2n+1}}{2^n n! (2n+1)} \right) \cdot \left(\sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} \right) \text{ или}$$

$$f(\xi) = \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \left(\sum_{n=0}^{+\infty} \frac{(-1)^n \xi^{2n+1}}{2^n n! (2n+1)} \right) \cdot \left(\sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} \right).$$

Тогда, используя правило Коши умножения рядов, получим

$$f(\xi) = \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \sum_{n=0}^{+\infty} \left(\sum_{k=0}^n \frac{(-1)^k}{2^k k! (2k+1) 2^{n-k} (n-k)!} \right) \xi^{2n+1} =$$

$$= \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \sum_{n=0}^{+\infty} \left(\sum_{k=0}^n \frac{(-1)^k}{k! (2k+1) (n-k)!} \right) \frac{\xi^{2n+1}}{2^n} =$$

$$= \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \sum_{n=0}^{+\infty} \left(\sum_{k=0}^n \frac{(-1)^k C_n^k}{(2k+1)n!} \right) \frac{\xi^{2n+1}}{2^n} =$$

$$= \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \sum_{n=0}^{+\infty} \left(\sum_{k=0}^n \frac{(-1)^k C_n^k}{(2k+1)} \right) \frac{\xi^{2n+1}}{2^n n!}.$$

Докажем, что $\sum_{k=0}^n \frac{(-1)^k C_n^k}{2k+1} = \frac{(2n)!!}{(2n+1)!!}$

Используя бином Ньютона, $(1-x^2)^n = \sum_{k=0}^n (-1)^k C_n^k x^{2k}$ и интегрируя, имеем

$$\int_0^1 (1-x^2)^n dx = \int_0^1 \sum_{k=0}^n (-1)^k C_n^k x^{2k} dx = \sum_{k=0}^n \frac{(-1)^k C_n^k x^{2k+1}}{2k+1} \Big|_0^1 = \sum_{k=0}^n \frac{(-1)^k C_n^k}{2k+1}.$$

Заметим, что с другой стороны

$$I_{m,n} = \int_0^1 x^m (1-x^2)^n dx = \int_0^1 \frac{(1-x^2)^n}{m+1} dx^{m+1} = \frac{x^{m+1} (1-x^2)^n}{m+1} \Big|_0^1 - \\ - \int_0^1 \frac{x^{m+1}}{m+1} d(1-x^2)^n = 2n \int_0^1 \frac{x^{m+2}}{m+1} (1-x^2)^{n-1} dx = \frac{2n}{m+1} I_{m+2,n-1}.$$

Тогда

$$I_{m,n} = \frac{(2n)!!}{\prod_{k=1}^n (m+2k-1)} I_{m+2n,0} \text{ и } I_{0,n} = \frac{(2n)!!}{\prod_{k=1}^n (2k-1)} I_{2n,0} = \frac{(2n)!!}{(2n-1)!!} I_{2n,0}.$$

Учитывая, что $I_{2n,0} = \int_0^1 x^{2n} dx = \frac{x^{2n+1}}{2n+1} \Big|_0^1 = \frac{1}{2n+1}$, имеем $I_{0,n} = \frac{(2n)!!}{(2n-1)!!} I_{2n,0} = \frac{(2n)!!}{(2n+1)!!}$ и

$$\sum_{k=0}^n \frac{(-1)^k C_n^k}{2k+1} = \int_0^1 (1-x^2)^n dx = I_{0,n} = \frac{(2n)!!}{(2n+1)!!}.$$

Заметим также, что значения $I_{m,n}$ связаны с бета-функцией

$$B(p, q) = \int_0^1 t^{p-1} (1-t)^{q-1} dt \text{ соотношением:}$$

$$I_{m,n} = \int_0^1 x^m (1-x^2)^n dx = \left| \begin{matrix} x^2 = t, x = \sqrt{t} \\ 2x dx = dt \end{matrix} \right| = \frac{1}{2} \int_0^1 t^{\frac{m-1}{2}} (1-t)^n dt = \frac{1}{2} B\left(\frac{m+1}{2}, n+1\right).$$

Тогда, используя свойства бета-функции $B(p, q)$ и гамма-функции $\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt$

– интегралы Эйлера первого и второго рода, получим

$$\begin{aligned} I_{0,n} &= \frac{1}{2} B\left(\frac{1}{2}, n+1\right) = \frac{\Gamma\left(\frac{1}{2}\right)\Gamma(n+1)}{2\Gamma\left(\frac{1}{2}+n+1\right)} = \frac{\Gamma\left(\frac{1}{2}\right)n!}{2\Gamma\left(\frac{1}{2}\right)\prod_{j=1}^{n+1}\left(\frac{1}{2}+n+1-j\right)} = \\ &= \frac{2^n n!}{2^{n+1} \prod_{j=1}^{n+1}\left(\frac{1}{2}+n+1-j\right)} = \frac{(2n)!!}{\prod_{j=1}^{n+1}(1+2n+2-2j)} = \frac{(2n)!!}{(2n+1)!!} \end{aligned}$$

Следовательно, $f(\xi) = \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \sum_{n=0}^{+\infty} \frac{\xi^{2n+1}}{(2n+1)!!}$.

Так как $\frac{d\theta_1}{d\xi} = f(\xi) = \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \sum_{n=0}^{+\infty} \frac{\xi^{2n+1}}{(2n+1)!!}$, то

$$\begin{aligned} \theta_1 &= \int \left(\sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n}}{2^n n!} - \sum_{n=0}^{+\infty} \frac{\xi^{2n+1}}{(2n+1)!!} \right) d\xi = \\ &= \sqrt{\frac{\pi}{2}} \sum_{n=0}^{+\infty} \frac{\xi^{2n+1}}{2^n n!(2n+1)} - \sum_{n=0}^{+\infty} \frac{\xi^{2n+2}}{(2n+1)!!(2n+2)} + C \end{aligned}$$

решение уравнения $\frac{d^2\theta_1}{d\xi^2} - \xi \frac{d\theta_1}{d\xi} = -1$.

Учитывая, начальное условие, $\theta_1(\xi)|_{\xi=\xi_*} = 0$, получаем, что

$$\theta_1(\xi) = S_1(\xi) - S_1(\xi_*), \tag{2}$$

где $S_1(\xi) = A(\xi) - B(\xi) = \sum_{n=0}^{+\infty} \frac{\sqrt{\frac{\pi}{2}} \xi^{2n+1}}{(2n)!!(2n+1)} - \sum_{n=0}^{+\infty} \frac{\xi^{2n+2}}{(2n+1)!!(2n+2)}$.

Общие члены этих рядов $a_n = a_n(\xi) = \sqrt{\frac{\pi}{2}} \cdot \frac{\xi^{2n+1}}{(2n)!!(2n+1)}$ и $b_n = b_n(\xi) = \frac{\xi^{2n+2}}{(2n+1)!!(2n+2)}$

удовлетворяют рекуррентным соотношениям $a_{n+1} = \frac{(2n+1)\xi^2}{(2n+2)(2n+3)} a_n$, $a_0 = \xi \sqrt{\frac{\pi}{2}}$ и

$b_{n+1} = \frac{(2n+2)\xi^2}{(2n+3)(2n+4)} b_n$, $b_0 = \frac{\xi^2}{2}$. Эти соотношения можно использовать для вы-

числения значений частичных сумм рядов $A(\xi)$, $B(\xi)$.

Отметим свойства рядов $A(\xi), B(\xi)$:

1. $a_n > 0, b_n > 0$, для любого $\xi > 0$.
2. $A(-\xi) = -A(\xi), B(-\xi) = B(\xi)$
3. $S_1(\xi) + S_1(-\xi) = -2B(\xi), S_1(\xi) - S_1(-\xi) = 2A(\xi)$.

Исследуем на сходимость ряды $A(\xi), B(\xi)$. Используя признак Д'Аламбера, имеем

$$\left| \frac{a_{n+1}}{a_n} \right| = \frac{\xi^2(2n+1)}{(2n+2)(2n+3)} < \frac{\xi^2}{2n+3} < q < 1, \quad \text{если} \quad n > \frac{\xi^2}{2q} - 1,5 \quad \text{и}$$

$$\frac{b_{n+1}}{b_n} = \frac{(2n+2)\xi^2}{(2n+3)(2n+4)} < \frac{\xi^2}{2n+4} < q < 1, \quad \text{если} \quad n > \frac{\xi^2}{2q} - 2.$$

Следовательно, для любого числа $q, 0 < q < 1$, существует натуральное число, на-

пример, $n_0 = \left\lceil \frac{\xi^2}{2q} \right\rceil$, такое, что для любого $n > n_0$ выполняются неравенства

$\left| \frac{a_{n+1}}{a_n} \right| < q, \frac{b_{n+1}}{b_n} < q$. Тогда остатки рядов $A(\xi), B(\xi)$ удовлетворяют неравенствам:

$$\left| \sum_{k=n}^{+\infty} a_k \right| \leq \frac{|a_n|}{1-q}, \quad \sum_{k=n}^{+\infty} b_k \leq \frac{b_n}{1-q}$$

и сходятся со скоростью, не меньшей, чем скорость бесконечно убывающей геометрической прогрессии со знаменателем q .

Таким образом, значения рядов $A(\xi), B(\xi)$, с заданной точностью $\varepsilon > 0$, можно получить, вычисляя n -ые частичные суммы этих рядов $\sum_{k=0}^{n-1} a_k, \sum_{k=0}^{n-1} b_k$, если выполняются неравенства:

$$|a_n| \leq \varepsilon(1-q), \quad b_n \leq \varepsilon(1-q), \quad \text{и} \quad n \geq n_0 = \left\lceil \frac{\xi^2}{2q} \right\rceil. \quad (3)$$

Так, например, для $q = 0,5$ неравенства принимают вид: $|a_n| \leq \varepsilon, b_n \leq \varepsilon$, и $n \geq n_0 = \lceil \xi^2 \rceil$.

Таким образом, точность 2ε вычисления значений $S_1(\xi), S_1(\xi_*)$ обеспечивается вычислением n -ых частичных сумм рядов $\sum_{k=0}^{n-1} a_k(\xi), \sum_{k=0}^{n-1} b_k(\xi)$ при выполнении условий (3), что гарантирует точность 4ε вычисления значения $\theta_1(\xi) = S_1(\xi) - S_1(\xi_*)$.

Учитывая следствие формулы Стирлинга: $n^n e^{-n} \sqrt{2\pi n} < n! < n^n \sqrt{2\pi n} e^{-n + \frac{1}{12n}}$, имеем

$$\begin{aligned} \left| \frac{a_n}{b_n} \right| &= \frac{\sqrt{\frac{\pi}{2}}(2n+1)!!(2n+2)}{|\xi|(2n)!!(2n+1)} = \frac{\sqrt{\frac{\pi}{2}}(2n)!(2n+2)}{|\xi|2^{2n}(n!)^2} > \\ &> \frac{\sqrt{\pi}}{|\xi|\sqrt{2}} \frac{(2n+2)(2n)^{2n} e^{-2n} 2\sqrt{\pi n}}{2^{2n} n^{2n} 2\pi n e^{-2n + \frac{1}{6n}}} = \frac{(n+1)\sqrt{2n}}{n e^{\frac{1}{6n}} |\xi|} > \frac{\sqrt{2n}}{e^{\frac{1}{6}} |\xi|} \geq 1, \end{aligned}$$

если $n \geq \frac{\sqrt[3]{e\xi^2}}{2}$. Следовательно, для $n \geq n_0 = \lceil \xi^2 \rceil$ выполняется неравенство $|a_n| \geq b_n$. Это неравенство позволяет упростить проверку условий (3) при вычислении n -ых частичных сумм рядов $\sum_{k=0}^{n-1} a_k(\xi)$ и $\sum_{k=0}^{n-1} b_k(\xi)$.

Результаты вычисления значений функции $\theta_1(\xi)$ с использованием соотношений (2) и условий (3) приведены в табл. 1 и на рис.1.

Таблица 1

ξ_*	ξ					
	0	1	2	3	4	5
-1	2,09	3,00	3,52	3,88	4,14	4,35
0		0,90	1,43	1,78	2,05	2,26
1			0,52	0,88	1,15	1,36
2				0,36	0,62	0,84

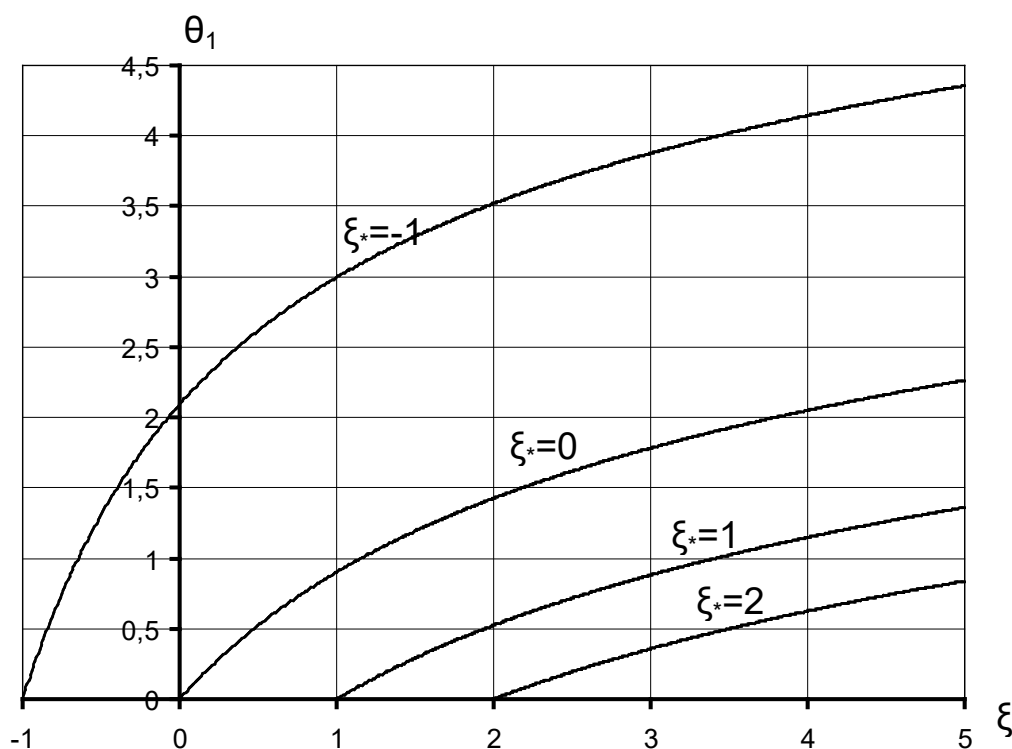


Рисунок 1

Литература

1. Найденов В.И., Швейкина В.И. Нелинейные модели колебаний речного стока / В.И. Найденов, В.И. Швейкина // Водные ресурсы. – Том 29, № 1. – М., 2002. – С. 62 – 67.
2. Волчек, А.А. Сравнительная оценка марковских и нелинейных моделей годового стока рек Беларуси / А.А. Волчек, С.И. Парфомук // Вестник БрГТУ. – Брест, 2006. – № 5 – С. 56-60.
3. Волчек, А.А. О решении одной стохастической модели многолетних колебаний речного стока / А.А. Волчек, И.И. Гладкий, Л.П. Махнист, С.И. Парфомук // Вестник БрГТУ. – Брест, 2008. – № 5 – С. 83-87.

УДК 517.925

ОБ ОДНОМ КЛАССЕ СИСТЕМ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Грицук Д.В., Климашевская И.Н.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

Рассмотрим автономную систему двух дифференциальных уравнений вида

$$\frac{dx}{dz} = P(x, y), \quad \frac{dy}{dz} = Q(x, y), \quad (1)$$

где x , y и z - комплексные переменные, P и Q - многочлены по x и y с постоянными коэффициентами. Пусть представления многочленов P и Q имеют вид

$$P(x, y) = \sum_{i=0}^{p_1} P_i(y)x^{p_1-i}, \quad Q(x, y) = \sum_{i=0}^{q_1} Q_i(y)x^{q_1-i},$$

где $P_i(y) = \sum_{j=0}^{l_i} p_{ij}y^{l_i-j}$, $Q_i(y) = \sum_{j=0}^{r_i} q_{ij}y^{r_i-j}$.

Обозначим $\max\{l_i\} = p_2$ ($i = 0, p_1$), $\max\{r_i\} = q_2$ ($i = 0, q_1$).

Задача: выделить класс систем дифференциальных уравнений вида (1), имеющих решение $(x(z), y(z))$ со свойством

$$x(z) \rightarrow \infty, \quad y(z) \rightarrow \infty \quad \text{при} \quad z \rightarrow z_0, \quad (2)$$

для функций $x(z)$ и $y(z)$ которого точка z_0 является полюсом, обычным или критическим.

Разделив первое уравнение системы (1) на второе, получим уравнение

$$\frac{dx}{dy} = \frac{\sum_{i=0}^{p_1} \left(\sum_{j=0}^{l_i} p_{ij}y^{l_i-j} \right) x^{p_1-i}}{\sum_{i=0}^{q_1} \left(\sum_{j=0}^{r_i} q_{ij}y^{r_i-j} \right) x^{q_1-i}}, \quad (3)$$

решение $x(y)$ которого обладает свойством $x(y) \rightarrow \infty$ при $y \rightarrow \infty$.

Сделав в уравнении (3) замену $y = \frac{1}{v}$, приведем его к виду

$$\frac{dx}{dv} = - \frac{\sum_{i=0}^{p_1} \left(\sum_{j=0}^{l_i} p_{ij}v^{p_2-l_i+j} \right) x^{p_1-i}}{\sum_{i=0}^{q_1} \left(\sum_{j=0}^{r_i} q_{ij}v^{q_2-r_i+j} \right) x^{q_1-i}} \cdot v^{q_2-p_2-2} \equiv \frac{M(x, v)}{N(x, v)}, \quad (4)$$

где M и N многочлены по x и v . Имеют место

Теорема 1 [1]. Если $p_1 \geq q_1 + 2$ и $q_2 \geq p_2 + 2$, то система (1) всегда имеет хотя бы одно решение со свойством (2), для обеих функций которого точка z_0 является полюсом, обычным или критическим. В окрестности точки z_0 это решение представляется в виде рядов

$$y = \sum_{i=0}^{\infty} B_i (z - z_0)^{i-s} \quad (B_0 \neq 0, m_1 \in Z_+), \quad x = \sum_{i=0}^{\infty} A_i (z - z_0)^{i-r} \quad (A_0 \neq 0), \quad (5)$$

где r, s, m_1 - целые положительные числа.

Теорема 2 [2]. Если выполнены условия

$$p_1 \geq q_1 + 2, \quad q_2 = p_2 + 2, \quad P_0(y) \text{ многочлен степени } p_2, \quad (6)$$

или

$$q_2 \geq p_2 + 2, \quad p_1 = q_1 + 2, \quad Q_0(y) \text{ многочлен степени } q_2, \quad (7)$$

или $p_1 \geq q_1 + 2, \quad q_2 \geq p_2 + 2$ и

$$l_i \leq l_0 + iR, \quad r_j \leq q_2 - (k - j)R, \quad \text{где } R = \frac{q_2 - l_0 - 1}{p_1 - q_1 - 1 + k}, \quad (8)$$

то система (1) имеет только алгеброидные решения со свойством (2).

Литература.

1. Грицук, Д.В. Существование полярных решений автономных полиномиальных систем двух дифференциальных уравнений / Грицук Д.В. // Материалы V международной научно-практической конференции молодых исследователей «Содружество наук. Барановичи - 2009» (21 – 22 мая 2009 года): В 2-х ч. – Барановичи: РИО БарГУ, 2009. – Ч. 2 – С. 71–73.

2. Грицук, Д.В. Условия алгеброидности решений автономных полиномиальных систем дифференциальных уравнений / Грицук Д.В., Климашевская И.Н. // Материалы республиканской научно-практической конференции «Современные проблемы математического моделирования и новые образовательные технологии в математике» (22 – 23 апреля 2009 года): В 2-х т. – Брест: альтернатива, 2009. – Т. 1 – С. 114–116.

УДК 519.23/.25

ОГРАНИЧЕНИЯ МЕТОДА НАИМЕНЬШИХ КВАДРАТОВ ПРИ ПОСТРОЕНИИ МОДЕЛЕЙ ЛИНЕЙНОЙ РЕГРЕССИИ

Дереченник С.С.

УО «Брестский государственный технический университет», г. Брест

Широкий круг задач обработки эмпирических данных имеет целью установление функциональной связи зависимой переменной (отклика) с одной или несколькими независимыми переменными (факторами). Например, в задаче простой линейной регрессии взаимосвязь между откликом y и фактором x находят в виде модели $y = ax + b$, при этом относительно имеющих наблюдений $(x_i, y_i), i = \overline{1, n}$ полагают, что $y_i = ax_i + b + \varepsilon_i$, при этом случайные величины (остатки) $\varepsilon_1, \dots, \varepsilon_n$, по условию гомоскедастичности, одинаково распределены и независимы.

Для подбора "наилучшей" прямой широко применяется известный метод наименьших квадратов ([1]), который заключается в такой оценке параметров (\hat{a}, \hat{b}) , которая минимизирует сумму квадратов величин e_i , называемых невязками:

$$\sum_i e_i^2 = \sum_{i=1}^n (y_i - ax_i - b)^2 \rightarrow \min_{a, b}. \quad (1)$$

Расчет уравнения простой линейной регрессии не представляет сложности. Некоторые проблемы возникают в случае построения моделей линейной регрессии с нелинейной связью факторов и отклика. Ряд таких функциональных форм используется, например, в табличном процессоре Microsoft Excel – для построения полиномиальной, логарифмической, степенной и экспоненциальной линий тренда.

Термин "линейный" в названии "линейный регрессионный анализ" указывает на линейность модели относительно коэффициентов (параметров), но не самого фактора / факторов. Некоторые авторы в связи с этим утверждают, что с выборкой $\{x_i\}$ можно проделать любые функциональные преобразования и включить преобразованный фактор в линейное уравнение [2]. Действительно, таким методом удобно линеаризовать, например, полиномы $y = a_1 x_i + a_2 x_i^2 + \dots + a_k x_i^k + b + \varepsilon_i$, сводя, тем самым, расчет параметров модели к стандартной задаче линейной регрессии. Важно при этом, что ошибка (остатки ε_i) уравнения остается аддитивной. Пример мультипликативного вхождения ошибки иллюстрируется следующей схемой линеаризации и обратного преобразования иной, например, степенной зависимости:

$$y = a x^b, \tag{2.1}$$

$$\lg y_i = \lg a + b \cdot \lg x_i + \varepsilon_i, \tag{2.2}$$

$$y_i = a \cdot x_i^b \cdot \varepsilon_i. \tag{2.3}$$

В подобных случаях регрессионные модели с линеаризованными функциональными зависимостями оказываются некорректными. Обычно рекомендуются два возможных выхода из такой ситуации.

1) Обратное преобразование вида (2.3) не выполняется, а в качестве окончательного и самодостаточного принимается линеаризованное уравнение (2.2), в рамках которого выполняется не только оценка параметров (\hat{a}, \hat{b}) , но и весь последующий анализ данных. В этом случае остатки ε_i аддитивны, но уже зависят от величины x , т.е. условие их гомоскедастичности не выполнено.

2) Преобразования (2.2) и (2.3) вообще не осуществляются, оценка параметров (\hat{a}, \hat{b}) выполняется с использованием итерационных вычислительных процедур нелинейной оптимизации, например, методом Ньютона-Рафсона. Данный подход реализуется в стандартных пакетах статистических программ.

Мы полагаем, что допустимость функциональных преобразований фактора x определяется также видом числовой шкалы, в которой представлены результаты его измерения. Известно, что операции логарифмирования, возведения в дробную степень, использования в качестве логарифма либо показателя степени неприменимы к результатам, представленным в шкале интервалов, разностей либо отношений (имеющим физическую единицу измерения), но возможны над результатами измерений в абсолютной шкале [2]. Поэтому отсутствие размерности у фактора x также является необходимым условием корректности линеаризации произвольной функциональной зависимости $y = f(x)$.

Линеаризация применяется, например, в задачах исследования вероятностных распределений экстремальных метеорологических величин, в частности, годовых максимумов снеговой нагрузки [3]. При этом эмпирическое распределение в области больших значений ("хвостовой" части распределения) проверяется на принадлежность к одному из характерных типов. Распределению Гумбеля (тип I) соответствует линейная аппроксимация нагрузки (отклика) вида $y = ax + b$, где $x = \ln(-\ln F(y))$. Данные действия допустимы, т.к. ошибка отклика остается аддитивной, а двойное логарифмирование выполняется над безразмерной вероятностью. Это сопровождается неравномерной группировкой точек наблюдений (x_i, y_i) .

На рисунке 1 представлены результаты расчета методом наименьших квадратов условной линейной модели регрессии с равномерным и неравномерным расположением точек наблюдений. Точки наблюдений соответствуют исходному уравнению $y_i = 4x_i + 5 + \varepsilon_i$, последовательные значения $\varepsilon_i \in [-5; 5]$, $i = 1, \dots, 24$ равномерно распределенной случайной ошибки идентичны для обоих вариантов. Усредненные по серии из 50 экспериментов (50 набором $\{\varepsilon_i\}$) результаты сведены в таблицу.

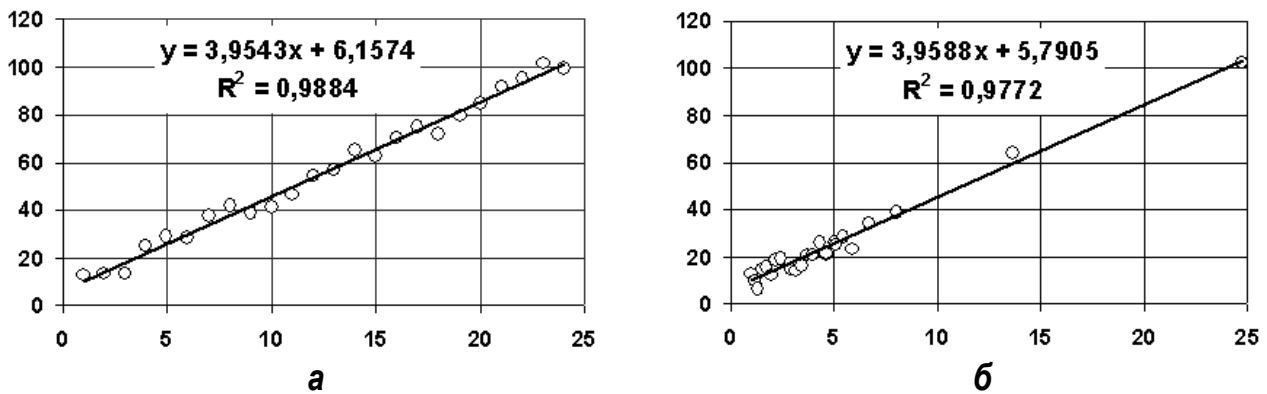


Рисунок – Регрессионные модели с равномерной (а) и неравномерной (б) группировкой точек наблюдений

Таблица 1 – Средние оценки параметров и коэффициент детерминации модели линейной регрессии $y = ax + b$

Модель с равномерной группировкой наблюдений		
параметр \hat{a}	параметр \hat{b}	R -квадрат
$3,998 \pm 0,003$	$5,16 \pm 0,72$	0,9897
Модель с неравномерной группировкой наблюдений		
параметр \hat{a}	параметр \hat{b}	R -квадрат
$3,995 \pm 0,007$	$5,16 \pm 0,35$	0,9802

Нами установлено, что при неравномерном расположении точек наблюдений, соответствие найденных оценок параметров модели исходным коэффициентам, в среднем, несколько ухудшается. Значение коэффициента детерминации R -квадрат при таком расположении закономерно (в каждом эксперименте) ниже. По-видимому, это связано с существенным смещением значений выборочных средних \bar{x} и \bar{y} , которые используются для расчета оценок (\hat{a}, \hat{b}) и коэффициента R^2 . Поэтому стандартный метод наименьших квадратов, основанный на "точечном" анализе невязок согласно соотношению (1), может оказаться не вполне корректным в случае неравномерного расположения точек наблюдений, например,

при нелинейном преобразовании шкалы фактора. Данный вывод косвенно подтверждается сведениями из раздела "О типах линий тренда" справочного руководства табличного процессора MS Excel, где указано (без каких-либо дополнительных пояснений), что "отображаемое вместе с линией тренда значение величины R-квадрат не является корректным".

Нами предлагается в качестве меры близости линии регрессии к эмпирическим точкам наблюдений использовать интегральную квадратичную оценку – функционал вида:

$$J = \int_{x_{\min}}^{x_{\max}} [y^*(x) - (ax + b)]^2 dx \rightarrow \min_{a, b}, \quad (3)$$

где $y^*(x)$ – некоторое кусочно-линейное приближение искомой функции, заданной точечными отсчетами (x_i, y_i) .

Подходящим приближением в общем случае может являться ломаная, а в задачах вероятностного типа – ступенчатая функция. В этих случаях функционал вычислим аналитически как для простой линейной регрессии, так и для некоторых ее нелинейных форм (например, экспоненциальной). Поэтому возможна также и аналитическая его минимизация, что позволит получить оценки параметров регрессии в замкнутой форме, без необходимости использования численных итерационных процедур. Аналогичный подход применим также для нахождения интегрального коэффициента детерминации.

Таким образом, минимизация интегральной квадратичной оценки при построении моделей регрессии нечувствительна к неравномерности расположения (группировки) точек наблюдения, не требует линеаризации некоторых нелинейных форм зависимости отклика и фактора, и, по-видимому, более корректна в отношении оценки коэффициента детерминации.

Литература

1. Корн, Г. Справочник по математике (для научных работников и инженеров) / Г. Корн, Т. Корн. – М.: Наука, 1978. – 832 с.
2. Шитиков, В.К. Количественная гидроэкология: методы системной идентификации / В.К. Шитиков, Г.С. Розенберг, Т.Д. Зинченко – Тольятти, ИЭВБ РАН, 2003. – 463 с.
3. Тур, В.В. Нормирование снеговых нагрузок для территории Республики Беларусь / Тур В.В., Валуев В.В., Дереченник С.С. [и др.] // Строительная наука и техника. – 2008. – № 2 (17). – С. 27–45.

УДК 517.512.2

АППРОКСИМАЦИОННЫЕ МЕТОДЫ ВЫЧИСЛЕНИЯ ИНТЕГРАЛОВ

Дыбко А.В., Дацьк В.Т.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

Пусть требуется вычислить определённый интеграл $\int_a^b f(x)dx$, где $f(x)$ есть некоторая заданная на промежутке $[a, b]$ непрерывная функция. Мы имеем много примеров вычисления подобных интегралов, либо с помощью первообразной, если она выражается в конечном виде, либо же – минуя первообразную – с помощью различных приёмов, большей частью искусственных. Нужно отметить, однако, всем этим исчерпывается лишь довольно узкий класс интегралов; за его пределами обычно прибегают к различным методам приближённого вычисления.

Дробление промежутка интегрирования.

При вычислении интеграла $\int_a^b f(x)dx$ можно поступить так. Разобьем сначала промежуток $[a, b]$ на некоторое число n равных промежутков

$$[x_0, x_1], [x_1, x_2], \dots, [x_{n-1}, x_n] \quad x_0 = a, x_n = b,$$

в связи с чем данный интеграл представится в виде суммы

$$\int_{x_0}^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \dots + \int_{x_{n-1}}^{x_n} f(x)dx. \quad (1)$$

Теперь же к каждому из этих промежутков применим параболическое интерполирование, т.е. станем вычислять интегралы (1) по одной из приближенных формул

$$\int_a^b f(x)dx = (b-a)f\left(\frac{a+b}{2}\right) \quad (2)$$

$$\int_a^b f(x)dx = (b-a)\frac{f(a)-f(b)}{2} \quad (3)$$

$$\int_a^b f(x)dx = \frac{b-a}{6} \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right] \quad (4)$$

Применим теперь к интегралу (1) формулу (4); при этом, для краткости, положим, как и выше,

$$f(x_i) = y_i, \quad \frac{x_i + x_{i+1}}{2} = x_{i+1/2}, \quad f(x_{i+1/2}) = y_{i+1/2}$$

Мы получим

$$\int_{x_0}^{x_1} f(x)dx = \frac{b-a}{6n}(y_0 + 4y_{1/2} + y_1),$$

$$\int_{x_1}^{x_2} f(x)dx = \frac{b-a}{6n}(y_1 + 4y_{3/2} + y_2),$$

.....
.....

$$\int_{x_{n-1}}^{x_n} f(x)dx = \frac{b-a}{6n}(y_{n-1} + 4y_{n-1/2} + y_n).$$

Наконец, складывая почленно эти равенства, придем к формуле:

$$\int_{x_{n-1}}^{x_n} f(x)dx = \frac{b-a}{6n} [(y_0 + y_n) + 2(y_1 + y_2 + \dots + y_{n-1}) + (y_{3/2} + 4y_{5/2} + y_{n-1/2})].$$

Она носит название формулы Симпсона; этой формулой пользуются для приближенного вычисления интегралов чаще, чем формулами прямоугольников и трапеций, ибо она - при той же затрате труда - дает обычно более точный результат.

Литература

1. Матвеев, Н.М. Дифференциальные уравнения / Н.М. Матвеев – Минск: Высшэйшая школа, 1976. – 368с. с ил.
2. Натансон, И.П. Теория функций вещественной переменной / И.П. Натансон – М.: Наука (Главная редакция физико-математической литературы), 1974. – 480 стр. с илл.
3. Самко, С.Г. Интегралы и производные дробного порядка и некоторые их приложения / С.Г. Самко, А.А. Килбас, О.И. Маричев – Минск: Наука и техника, 1987. – 688 с.

УДК 517

ЧИСЛЕННАЯ РАСЧЕТНАЯ МОДЕЛЬ ДЛЯ ОПРЕДЕЛЕНИЯ ОСНОВНЫХ ХАРАКТЕРИСТИК РАЗРАБОТКИ ГАЗОВЫХ ЗАЛЕЖЕЙ С ПОЛЗУЧЕЙ СРЕДОЙ

Б.З.Казымов, К.К.Насирова

*Институт геологии Национальной Академии Наук Азербайджана,
г. Баку, Азербайджан*

Горные породы глубокозалегающих месторождений нефти и газа, находящиеся в процессе разработки, могут подвергаться сильным, часто неупругим – релаксационным и ползучим деформациям. Учет этих явлений позволит существенным образом повысить точность и надежность гидродинамических расчетов по прогнозированию показателей разработки указанного типа месторождений.

В связи с этим настоящая работа посвящена численному моделированию процесса неустановившейся фильтрации реального газа к центральной скважине в залежах с ползучей средой. Исследовалась краевая задача в следующей математической постановке: требуется решить нелинейное дифференциальное уравнение

$$\frac{1}{r} \frac{\partial}{\partial r} \left(\frac{p}{\mu(p)z(p)} \frac{\partial p}{\partial r} \right) = \frac{1}{k} \frac{\partial}{\partial t} \frac{mp}{z(p)} \tag{1}$$

совместно с законом релаксации пористости в ползучей среде

$$m = m_0 \left[1 + \beta_n (p - p_0) + m_1 \int_0^t e^{-\gamma_m(t-\tau)} (p - p_0) d\tau \right] \tag{2}$$

при соответствующих начальных и граничных условиях

$$p(r,0) = p_0, \quad m(r,0) = m_0$$

$$\begin{cases} \left. \frac{2\pi r h k \beta p}{\mu(p)z(p) p_{am}} \frac{\partial p}{\partial r} \right|_{r=r_c} = q(t) \\ \left. \frac{\partial p}{\partial r} \right|_{r=r_k} = 0 \end{cases} \tag{3}$$

где p – текущее пластовое давление; p_0 – начальное пластовое давление; q – дебит газовой скважины; m, k, h – соотве пористость, проницаемость и толщина пласта; m_0 – начальная пористость пласта; μ – вязкость тственно, газа; z – коэффициент сверхсжимаемости газа;

p_{atm} - атмосферное давление; $\gamma_m = 1/\tau_m \tau_m$ - время релаксации пористости; β_n - коэффициент сжимаемости пород пласта; β - температурная поправка для газа; m_1 - параметр ползучести; r_c - радиус скважины; r_k - радиус залежи; r - радиальная координата; t - время; $r_c \leq r \leq r_k, t \geq 0$.

Численное решение задачи (1)-(3) получено с применением метода прогонки с итерационным уточнением нелинейных коэффициентов [1,2]. В результате впервые разработана численная расчетная модель для прогнозирования основных характеристик разработки газовой залежи с ползучей средой, таких как пластовое и забойное давления, пористости пласта у стенки скважины и на контуре залежи и т.д.

Литература

1. Азиз, Х. Математическое моделирование пластовых систем: пер. с англ. / Х. Азиз, Э. Сеттари – Москва: Изд. «Недра», 1982. – 407 с.
2. Закиров С.Н. Разработка газовых, газоконденсатных и нефтегазоконденсатных месторождений / С.Н. Закиров – Москва: изд. «Струна», 1998. – 628 с.

УДК 517.948

О ВОССТАНОВЛЕНИИ СЕТОЧНОГО РЕШЕНИЯ ЛИНЕЙНОЙ ДИФФЕРЕНЦИАЛЬНОЙ ЗАДАЧИ ВТОРОГО ПОРЯДКА

Калач А.В., Шепшелей Д.В.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель - Мадорский В.М., кандидат физ.-мат. наук, доцент*

Рассмотрим задачу Коши с известным решением $y(x) = x \ln(x)$:

$$y''(x) + y'(x) - \frac{1}{x} y(x) = \frac{x+1}{x}, \quad (1)$$

$$\begin{aligned} y(0.5) &= \ln(0.5) / 2, \\ y'(0.5) &= \ln(0.5) + 1, \\ 0.5 &\leq x \leq 1. \end{aligned} \quad (2)$$

Будем решать задачу (1)-(2) методом Рунге-Кутты IV порядка точности.

$$\begin{aligned} k_1 &= f(x_i, y_i), \\ k_2 &= f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} k_1\right), \\ k_3 &= f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} k_2\right), \\ k_4 &= f(x_i + h, y_i + h k_3), \\ \Delta y_i &= \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4), \\ y_{i+1} &= y_i + \Delta y_i. \end{aligned}$$

Предварительно редуцируем дифференциальную задачу II порядка в систему 2-х дифференциальных задач I порядка:

$$\left\{ \begin{array}{l} y'(x) = z, \\ z'(x) = \frac{x+1}{x} + \frac{1}{x}y(x) - z \end{array} \right\}.$$

При численном решении этой системы нами использовались следующие параметры: $E=10^{-15} - 10^{-17}$ – точность применения принципа Рунге n – количества корней полинома Чебышева I рода, расположенных на отрезке $[0.5;1]$. m – количество точек разбиения отрезка $[0.5;1]$, не совпадающих с корнями полинома Чебышева на этом отрезке. После нахождения приближенного значения решения в корнях полинома Чебышева I рода произведена процедура восстановления каркаса приближенного решения по формулам:

$$T_n(t) = \cos(n \cdot \arccost t),$$

$$t_j = \cos \frac{(2j-1)\pi}{2n}, j = \overline{1, n},$$

$$x_j = \frac{b-a}{2}t_j + \frac{b+a}{2},$$

$$P_m(x) = \frac{c_0}{2} + \sum_{k=1}^m c_k T_k\left(\frac{2x-b-a}{b-a}\right),$$

$$c_k = \frac{2}{n} \sum_{j=1}^n f(x_j) T_k\left(\frac{2x_j-b-a}{b-a}\right),$$

$$k = \overline{0, m}, m+1 \leq n.$$

Восстановленное приближенное решение $\tilde{y}(x)$ тестировалось в некоторой совокупности точек, не совпадающих с корнями полинома Чебышева. В дальнейшем этот результат сравнивался с точным решением $y(x)$ в этих точках. Результат тестирования следующий:

E	n – число корней полинома Чебышева			m – число точек разбиения промежутка			$\ y(x) - \tilde{y}(x)\ $
	10	20	30	20	40	60	
1e-14	+			+			6,09065E-11
1e-14		+			+		6,16928E-14
1e-14			+			+	4,99294E-14
1e-15	+			+			6,09064E-11
1e-15		+			+		1,37079E-14
1e-15			+			+	1,10403E-14
1e-17	+			+			6,09062E-11
1e-17		+			+		4,51479E-15
1e-17			+			+	3,19075E-15

Вывод: вычислительный эксперимент показал, что $\|y(x) - \tilde{y}(x)\|$ будет минимальной, если степень полинома Чебышева будет равна $n-1$, точность аппроксимации приближенного решения существенно зависит от точности применения принципа Рунге.

Литература

1. Мадорский, В.М. Квазиньютоновские процессы для решения нелинейных уравнений / В.М. Мадорский. – Брест: Изд-во БрГУ. – 2005. – 185 с.
2. Вержбицкий В.М. Основы численных методов / В.М. Вержбицкий. – Минск: Высшая школа. – 2002. – 840 с.

УДК 517.928.4.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОДНОГО БИОЛОГИЧЕСКОГО ПРОЦЕССА**Кожух И.Г., Корделюк О.И.***УО «Брестский государственный университет им. А.С.Пушкина», г. Брест*

Прежде чем приступить к конструированию модели прикладного характера, необходимо чётко представить, на какие вопросы можно получить ответ с помощью данной модели. Чем меньшим количеством уравнений можно обойтись при описании явлений, тем лучше модель. Только тогда, когда число независимых переменных не больше двух-трёх, можно уверенно разобраться в решении задачи и выявить влияние различных параметров. При увеличении числа уравнений результаты становятся необозримыми, и основное преимущество изучения модели вместо самого явления пропадает.

Рассмотрим модель процесса роста клеток в проточном культиваторе. Принцип действия проточного культиватора очень прост. В сосуд объёма V см³ (культиватор) со скоростью v см/с через специальное отверстие поступает раствор питательного вещества известной концентрации. В культиваторе он тщательно перемешивается с имеющимся раствором, содержащим биомассу X , то же питательное вещество заданной концентрации и некоторое количество продуктов жизнедеятельности клеток. Данный раствор с такой же скоростью v см/с выливается из сосуда через второе отверстие.

Основное замечательное свойство проточного метода-это осуществление устойчивого состояния равновесия. Через некоторое время после включения протока в сосуде устанавливаются постоянные, не зависящие от времени концентрации биомассы, вещества и продуктов жизнедеятельности. Это означает, что прирост биомассы из-за размножения в точности равен убыли из-за протока. В описанной ситуации состояние равновесия устойчиво. Пусть случайным образом изменилась концентрация биомассы в сосуде, тогда вырастет потребление питательного вещества и его концентрация станет меньше равновесной, что, в свою очередь, вызовет уменьшение скорости роста биомассы, вследствие чего произойдет уменьшение биомассы до равновесного состояния.

Попытаемся составить модель роста биомассы в культиваторе. Если он непроточный, то кинетика роста биомассы в нем описывается дифференциальным уравнением.

$$\frac{dx}{dt} = \mu X$$

Но одно это уравнение ещё не определяет всего процесса. Значит, более адекватно состояние процесса можно описать системой уравнений.

Рассмотрим простейшую систему уравнений проточного культиватора

$$\frac{dx}{dt} = \frac{\mu y}{K_y + y} x - Dx,$$

$$\frac{dy}{dt} = -\alpha \frac{\mu y}{K_y + y} x + D(y_0 - y),$$

(1)

где μ скорость роста, а D - коэффициент разбавления.

При заданном значении коэффициента D в культиваторе устанавливается постоянное значение удельной скорости роста μ .

Исследуем стационарные режимы системы (1) с целью определить, какие скорости потока возможны для нормальной работы культиватора. Для этого в уравнениях системы (1) положим производные по времени равными нулю. Получим два алгебраических уравнения

$$\alpha \bar{x} + \bar{y} = y_0, \quad (2)$$

$$\bar{x} \left(\frac{\mu \bar{y}}{K_y - \bar{y}} - D \right) = 0 \quad (3)$$

Уравнение (2) дает линейную связь между стационарными концентрациями \bar{x} и \bar{y} и должно выполняться всегда. Это условие постоянства экономического коэффициента $Y = \frac{1}{\alpha}$. Этот коэффициент показывает, какой процент пищи превращается непосредственно

в клеточную массу. Уравнению (3) соответствуют две возможности: либо

$$\bar{x} = \bar{x}_a = 0, \quad (4a)$$

а тогда из (2)

$$\bar{y} = \bar{y}_a = y_0, \quad (4b)$$

либо $\bar{x} \neq 0$, тогда

$$\bar{y} = \bar{y}_b = \frac{K_y D}{\mu - D} \quad (5a)$$

$$\bar{x} = \bar{x}_b = \frac{1}{\alpha} (y_0 - \bar{y}_b) = \frac{1}{\alpha} \cdot \frac{\mu y_0 - D(K_y + y_0)}{\mu - D}, \quad (5b)$$

где y_0 - постоянное значение, к которому с течением времени стремится суммарная концентрация u углерода ($u = x + y$).

Начертив графики зависимости \bar{x}_a , \bar{y}_a , \bar{x}_b , \bar{y}_b от коэффициента разбавления D ,

можно показать, что при $D > D_{\text{êđ}}$, где $D_{\text{êđ}} = \frac{\mu y_0}{K_y + y_0}$ (6) и \bar{x}_b становится отрица-

тельным. Это значит, что решения (5a) и (5b) перестанут иметь смысл, поскольку концентрации не могут быть отрицательными. А тогда в этой области значения параметра D может существовать только одно нулевое решение.

Вывод: при $D > D_{\text{êđ}}$, наступает режим вымывания, т.е. скорость потока настолько велика, что прирост биомассы не может компенсировать ее отток.

При скоростях потока, меньших $D_{\text{êđ}}$, имеют смысл оба решения (4a), (4b) и (5a), (5b). Для исследования вопроса, в каком режиме работает культиватор, необходимо проанализировать устойчивость обоих возможных режимов. Можно показать, что нулевое решение неустойчиво при $D < D_{\text{êđ}}$. Это значит, что если не поместить в культиватор никакой "затравки" микроорганизмов, то биомасса в нем не вырастет.

Рассмотрим далее случай $D < D_{\text{êđ}}$, исследуя вопрос об оптимальных режимах работы культиватора. Производительностью культиватора называют количество биомассы, снимаемой за единицу времени с единицы объема установки.

$$w = \frac{\Delta m}{V} = Dx. \quad (7)$$

Из (7) следует, что при $D \rightarrow 0$, производительность тоже стремится к нулю. Она равна нулю и тогда, когда происходит вымывание биомассы, т.е. $x = 0$. Нетрудно показать, что при определённых значениях $D = D_0$ производительность достигает максимума, что произойдет тогда, когда

$$D_0 = \mu \left(1 - \sqrt{\frac{K_y}{y_0 + K_y}} \right), w = \frac{\mu}{\alpha} \left(\sqrt{y_0 + K_y} - \sqrt{K_y} \right) \quad (8)$$

Вернемся опять к системе(1) и введем там безразмерные переменные и параметры:

$$\tilde{x} = \frac{\alpha x}{y_0}, \quad \tilde{y} = \frac{y}{y_0}, \quad \tilde{t} = t\mu, \quad v = \frac{D}{\mu}, \quad K = \frac{K_y}{y_0}.$$

Сохранив за переменными обозначения x, y, t вместо системы (1) будем иметь:

$$\frac{dx}{dt} = \frac{xy}{K+y} - \sigma = P(x, y), \quad (9)$$

$$\frac{dy}{dt} = -\frac{xy}{K+y} + \sigma(1-y) = Q(x, y)$$

Уравнение изоклины вертикальных касательных $P(x, y) = 0$ определяет на фазовой плоскости две прямые: $x = 0$ $y = \frac{\sigma K}{1-\sigma} = y_c$. Изоклина горизонтальных касательных $Q(x, y) = 0$ является гиперболой, одна из асимптот которой - ось x , а кривая $Q(x, y)$ пересекает ось y в точке $y = 1$ при любом значении σ . Особыми точками системы (9) являются:

$$x_a = 0, \quad y_a = 1; \quad (10)$$

$$x_b = \frac{1-\sigma(K+1)}{1-\sigma}, \quad y_b = \frac{K-\sigma}{1-\sigma}, \quad (11)$$

Причем, при $\sigma \neq 0$ обе эти точки лежат на прямой $x + y = 1$, и при увеличении скорости протока σ вторая точка перемещается влево вверх вдоль прямой $x + y = 1$. При $\sigma = \sigma_{кр}$ обе точки сливаются в одну, а при $\sigma > \sigma_{кр}$ такой режим не имеет физического смысла, т.к. x и y не могут быть отрицательными.

Исследовав характер и устойчивость особых точек, приходим к выводу:

а) точка $a(\bar{x}_a = 0, \bar{y}_a = \mu)$ при $\sigma < \sigma_{кр}$ является седлом, а при $\sigma > \sigma_{кр}$ - устойчивым узлом.

б) точка $b(\bar{x}_b, \bar{y}_b)$ существует лишь при $\sigma < \sigma_{кр} = \frac{1}{K+1}$ и является устойчивым узлом.

Выводы: а) процессы установления в соответствующей системе всегда носят апериодический характер.

б) качественное исследование системы (9) на фазовой плоскости позволяет определить положение и характер устойчивости особых точек, найти критические значения, а также и условия сосуществования.

Литература

1. Чернавский, Д.С. Колебательные процессы в биологических и химических системах / Д.С. Чернавский, Л. Н. Григоров, М.С. Полякова. – М.: Наука, 1967.

2. Андронов, А.А. Теория колебаний / А.А. Андронов, А.А. Витт, С.Э. Хайкин. – М.: Физматгиз, 1959.

УДК 517.928.4.

МЕТОД МЕДЛЕННО МЕНЯЮЩИХСЯ АМПЛИТУД В ИССЛЕДОВАНИИ ДИНАМИЧЕСКИХ СИСТЕМ

Кожух И.Г., Касперович Ю.А.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

Рассмотрим один из приближенных методов исследования некоторого класса динамических систем, имеющих предельный цикл, т.е. автоколебательных систем.

Пусть дана динамическая система

$$\frac{dx}{dt} = P(x, y), \quad \frac{dy}{dt} = Q(x, y), \quad (1)$$

где $P(x, y)$, и $Q(x, y)$ – заданные аналитические функции. Поместим начало координат в особую точку $(0, 0)$ системы (1), тогда она примет вид:

$$\frac{dx}{dt} = a_{11}x + a_{12}y + \mu f_1(x, y), \quad (2)$$

$$\frac{dy}{dt} = a_{21}x + a_{22}y + \mu f_2(x, y).$$

Система (2) получается из системы (1) в результате разложения функций $P(x, y)$ и $Q(x, y)$ в ряд Тейлора.

Известный французский ученый А.Пуанкаре в начале прошлого века показал, что в одном важном частном случае можно получить приближенное решение $x(t), y(t)$ такой системы. Этот случай относится к автоколебательным системам, имеющим предельный цикл.

Сформулированная выше задача может быть решена при выполнении двух условий.

1. Нелинейные функции $\mu f_1(x, y)$ и $\mu f_2(x, y)$ малы по сравнению с коэффициентами a_{ij} . Число μ в этом случае называют малым параметром $\mu \ll 1$, a_{ij} и f – одного порядка.

2. Если $\mu = 0$, то система (2) переходит в консервативную колебательную систему с особой точкой типа “центр”.

Рассмотрим динамическую систему

$$\frac{dx}{dt} = y, \quad \frac{dy}{dt} = -\omega^2 x + \mu f(x, y), \quad (3)$$

с помощью которой проиллюстрируем метод медленно меняющихся амплитуд. Система (3) является частным случаем системы (2) и в ряде случаев может быть получена из нее заменой переменных. Очевидно, система (3) эквивалентна одному дифференциальному уравнению второго порядка

$$\frac{d^2x}{dt^2} + \omega^2 x = \mu f\left(x, \frac{dx}{dt}\right) \quad (4)$$

При $\mu = 0$ траектории системы (3) на фазовой плоскости становятся концентрическими эллипсами, а особая точка (0,0) превращается в центр. А тогда решением уравнения (4) является гармоническая функция

$$x = A_0 \cos(\omega t + \varphi_0), \quad (5)$$

в которой постоянные A_0 и φ_0 определяются начальными условиями.

При $\mu \neq 0$, но $\mu \ll \omega$ система (3) или равносильное ей уравнение (4) описывают колебательные системы, близкие к гармоническим, а при $\mu \gg \omega$ колебательная система будет далека от гармонической.

Рассмотрим подробнее метод медленно меняющихся амплитуд, идея которого была предложена физиком Ван-дер-Полем. Метод основан на следующих рассуждениях. Если автоколебания близки к синусоидальным, а при отклонении от устойчивого предельного цикла изображающая точка снова приближается к нему достаточно медленно, то решение (5) можно представить в виде

$$x = A(t) \cos(\omega t + \varphi(t)), \quad (6)$$

где $A(t)$ и $\varphi(t)$ медленно меняющиеся функции времени. Это значит, что амплитуда колебаний мало изменяется за период и отставание или опережение по фазе за период мало по сравнению с 2π .

Продифференцируем выражение (6) по времени как сложную функцию:

$$\dot{x} = \dot{A} \cos(\omega t + \varphi) - A \dot{\varphi} \sin(\omega t + \varphi) - \omega A \sin(\omega t + \varphi). \quad (7)$$

Видно, что первые два слагаемые в правой части (7) содержат производные от медленно меняющихся функций $A(t)$ и $\varphi(t)$, имеющих порядок малости μ . Если бы $\mu = 0$, то A и φ не зависели бы от времени и

$$\dot{x} = -\omega A \sin(\omega t + \varphi), \quad (8)$$

Потребуем, чтобы равенство (8) выполнялось всегда, т.е. чтобы

$$\dot{A} \cos(\omega t + \varphi) - A \dot{\varphi} \sin(\omega t + \varphi) = 0 \quad (9)$$

Вторая производная \ddot{x} находится после дифференцирования (8) по времени

$$\ddot{x} = -\omega \dot{A} \sin(\omega t + \varphi) - \omega A \dot{\varphi} \cos(\omega t + \varphi) - \omega^2 A \cos(\omega t + \varphi) \quad (10)$$

Подставим теперь (6), (8), (10) в исходное уравнение (4), получим

$$\begin{aligned} & -\omega A \sin(\omega t + \varphi) - \omega A \dot{\varphi} \cos(\omega t + \varphi) = \\ & = \mu f[A \cos(\omega t + \varphi), -\omega A \sin(\omega t + \varphi)] \end{aligned} \quad (11)$$

Уравнение (9) и (11) представляют собой пока точную систему относительно переменных A и φ . Может показаться, что на вид система (9) и (11) сложнее, чем исходное уравнение (4) или система (3). Однако, как будет показано дальше, уравнения для A и φ можно упростить, что невозможно для исходной системы.

Разрешим (9) и (11) относительно \dot{A} и $\dot{\varphi}$. Из (9) имеем

$$\dot{\varphi} = \frac{\dot{A}}{A} \cdot \frac{\cos(\omega t + \varphi)}{\sin(\omega t + \varphi)} \quad (12)$$

Подставим (12) в (11) и умножим правую и левую части на $-\frac{1}{\omega} \sin(\omega t + \varphi)$. Получим

$$\dot{A} = -\frac{\mu}{\omega} \cdot \sin(\omega t + \varphi) \cdot f[A \cos(\omega t + \varphi), -\omega A \sin(\omega t + \varphi)]. \quad (13)$$

Подставив далее (13) в (12), будем иметь:

$$\dot{\varphi} = -\frac{\mu}{\omega} \cdot \frac{1}{A} \cos(\omega t + \varphi) \cdot f[A \cos(\omega t + \varphi), -\omega A \sin(\omega t + \varphi)]. \quad (14)$$

Воспользуемся медленностью функций $A(t)$ и $\varphi(t)$. Справа в уравнения (13), (14) входят быстроменяющиеся функции времени $\sin(\omega t + \varphi)$ и $\cos(\omega t + \varphi)$. Усредним их за период $\frac{2\pi}{\omega} = T$, а медленные функции $A, \varphi, \dot{A}, \dot{\varphi}$ приближенно заменим их средними за период значениями:

$$\dot{A} = \psi(A), \quad \dot{\varphi} = \Phi(A), \quad (15)$$

где

$$\left. \begin{aligned} \psi(A) &= -\frac{\mu}{\omega} \cdot \frac{1}{T} \int_0^T \sin(\omega t + \varphi) \cdot f[A \cos(\omega t + \varphi), -\omega A \sin(\omega t + \varphi)] dt, \\ \Phi(A) &= -\frac{\mu}{\omega} \cdot \frac{1}{T} \int_0^T \cos(\omega t + \varphi) \cdot f[A \cos(\omega t + \varphi), -\omega A \sin(\omega t + \varphi)] dt. \end{aligned} \right\} \quad (16)$$

Нетрудно видеть, что усредненные по периоду функции $\psi(A)$ и $\Phi(A)$ не зависят от медленной фазы $\varphi(t)$, а являются только функциями амплитуды. В самом деле, в (16) вместо переменной интегрирования t можно ввести новую переменную $z = \omega t + \varphi$, тогда $dz = \omega dt + d\varphi \approx \omega dt$ и функция $\psi(A)$ запишется в виде

$$\psi(A) = \frac{\mu}{\omega} \cdot \frac{1}{\omega \cdot t} \int_0^{\omega T} \sin z \cdot f[A \cos z, -A \sin z] dz,$$

где φ уже явно не входит. Аналогичные рассуждения можно провести относительно функции $\Phi(A)$.

И в заключение отметим, что кажущиеся на первый взгляд свойства амплитуды и фазы конкретной системы носят частный характер, однако практика и развивающаяся теория показывают, что все квазигармонические колебательные системы, имеющие один предельный цикл, мало чем качественно отличаются от задачи о ламповом генераторе, разрешенной Ван-дер-Подем.

Литература

1. Андронов, А.А. Теория колебаний / А.А. Андронов, А.А. Витт, С.Э. Хайкин. – М.: Физматгиз, 1959.
2. Андронов, А.А. Качественная теория динамических систем / А.А. Андронов [и др.] – М.: Наука, 1966.
3. Андронов, А.А. Теория бифуркаций динамических систем на плоскости / А.А. Андронов [и др.] – М.: Наука, 1967.
4. Степанов, В.В. Курс дифференциальных уравнений / В.В. Степанов. – М.-Л.: Гостехиздат, 1950.
5. Стрелков, С.П. Введение в теорию колебаний / С.П. Стрелков. – М.: Наука. 1964.

УДК 517

СТАЦИОНАРНОЕ РАСПРЕДЕЛЕНИЕ СЕТИ С ГРУППОВЫМИ ПЕРЕМЕЩЕНИЯМИ В ОБОБЩЕННОЙ МУЛЬТИПЛИКАТИВНОЙ ФОРМЕ

Коробейникова Е.В.

*УО «Гомельский государственный технический университет
имени П.О.Сухого», г. Гомель*

Введение. В большинстве задач массового обслуживания стационарное распределение в мультипликативной форме находится для сети, узлы в которой квазиобратимы. Т.е. происходит стандартная процедура склеивания узлов по теореме Келли, если узлы в сети будут квазиобратимыми, то стационарное распределение будет иметь форму произведения. В данной работе автору удалось выявить условия существования стационарного распределения в сети, состоящей из двух узлов, в форме смещенного геометрического произведения для неквазиобратимых узлов. Это несколько иная форма произведения, чем та, что получается при склеивании в сеть квазиобратимых узлов, при

которой множители в форме произведения не зависят от состояния других узлов. В полученном случае, множители в форме произведения зависят от состояния других узлов. Такую форму произведения назовем обобщенной мультипликативной формой.

1. Модель однолинейного узла. Рассмотрим однолинейную систему массового обслуживания с групповым поступлением и групповым обслуживанием. В систему поступает стационарный пуассоновский поток сообщений с интенсивностью λ . В момент поступления сообщения мгновенно формируется группа заявок случайного размера X_n (n -номер n -го по счету поступившего сообщения). Эта группа заявок присоединяется к очереди, если в системе есть другие заявки, в противном случае из заявок этой группы формируется группа заявок, которая сразу начинает обслуживаться. Механизм формирования требуемой для обслуживания группы точно такой, как описанный ниже механизм формирования группы на обслуживание после окончания обслуживания очередной группы. Предполагается, что $\{X_n, n = 1, 2, \dots\}$ - последовательность независимых неотрицательных одинаково распределенных целочисленных случайных величин с конечным математическим ожиданием m_A и вероятностными значениями

$a(k) = P\{X_n = k\}, \quad k = 1, 2, \dots$. Пусть $\tilde{A}(z) = \sum_{k=1}^{\infty} a(k)z^k$ - производящая функция X_n . В момент окончания обслуживания очередной группы на обслуживание выбирается группа заявок случайного размера Y_n , которая обслуживается целиком, при этом обслуживание - экспоненциальное с интенсивностью μ (n - номер n -й по счету обслуженной группы). Если в момент окончания обслуживания очередной группы размер требуемой для обслуживания группы строго больше числа оставшихся заявок в системе, то на обслуживание выбирается некомплектная группа из всех оставшихся в системе заявок. Предполагается, что $\{Y_n, n = 1, 2, \dots\}$ - последовательность независимых неотрицательных одинаково распределенных целочисленных случайных величин с конечным математическим ожиданием m_B и вероятностями значений $b(k) = P\{Y_n = k\}, \quad k = 1, 2, \dots$. Пусть также $\tilde{B}(z) = \sum_{k=1}^{\infty} b(k)z^k$ - производящая функция Y_n . Процессы поступления и обслуживания предполагаются независимыми. Размеры поступающих групп и размеры выбираемых на обслуживание групп также независимы.

Пусть $n(t)$ - число заявок в системе в момент t . Семейство случайных величин $\{n(t)\}$ есть цепь Маркова с непрерывным временем в пространстве состояний $Z_+ = \{0, 1, \dots\}$. Интенсивности ее перехода есть

$$\begin{aligned} q(n, n+k) &= \lambda a(k), & k \geq 1, n \geq 0, \\ q(n, n-k) &= \mu b(k), & n > k \geq 1, \\ q(n, 0) &= \mu \bar{B}(n) = \mu(1 - b(1) - b(2) - \dots - b(n-1)), & n \geq 1. \end{aligned} \tag{1.1}$$

В [1-3] установлено, что необходимым и достаточным условием эргодичности процесса $n(t)$ является условие

$$\lambda \cdot m_A < \mu \cdot m_B, \tag{1.2}$$

которое будет предполагаться всюду далее выполненным.

Уравнения равновесия для стационарных вероятностей процесса $X(t)$ можно записать в виде

$$\rho P(0) = \sum_{k=1}^{\infty} P(k) \bar{B}(k), \quad (1.3)$$

$$(1 + \rho)P(n) = \sum_{k=1}^{\infty} P(n+k)b(k) + \rho \sum_{k=1}^n P(n-k)a(k), \quad n \geq 1, \quad \text{где } \rho = \frac{\lambda}{\mu} \quad (1.4)$$

Решение (1.3), (1.4) будем искать в виде смещенного геометрического распределения $\{P(n), n = 0, 1, \dots\}$, где

$$P(0) = P_0, \quad P(n) = (1 - P_0)(1 - c)^{n-1}, \quad n = 1, 2, \dots, \quad P_0, c \in (0, 1). \quad (1.5)$$

Подставляя (1.5) в (1.3), получим $\rho P_0 = (1 - P_0)(1 - \bar{B}(c))$, откуда

$$\bar{B}(c) = \frac{1 - (1 + \rho)P_0}{1 - P_0}. \quad (1.6)$$

Известно [4], что для того, чтобы функция $\bar{B}(z) = \sum_{n=1}^{\infty} b(n)z^n$ являлась производящей функцией вероятностного распределения на $N = \{1, 2, \dots\}$, необходимо и достаточно, чтобы она была абсолютно монотонной на $(0, 1)$ (т.е. в любой точке $(0, 1)$ ее производные любых порядков были неотрицательными) и $\bar{B}(0) = 0$, $\bar{B}(1) = 1$. В частности, тогда она строго возрастает на $[0, 1]$ и нестрого выпукла вниз.

Рассмотрим уравнение

$$\bar{B}(c) = \alpha. \quad (1.7)$$

В силу вышесказанного $\alpha \in (0, 1)$, если и только если $c \in (0, 1)$. В силу строго возрастания и нестрогой выпуклости $\bar{B}(c)$, очевидно, $c \geq \alpha$. Таким образом, имеет место

Лемма 1. *Если $\alpha \in (0, 1)$, то уравнение (1.7) имеет единственный корень c . Этот корень находится в промежутке $[\alpha, 1)$.*

По лемме 1, если решение уравнений равновесия (1.3), (1.4) в форме (1.5) существует, то необходимо выполняется $P_0 < \frac{1}{1 + \rho}$. Согласно (1.5) $P(n+k) = P(n)c^k$. Умножая (1.4) на $(z)^n$ и складывая по всем $n = 1, 2, \dots$, имеем

$$(\lambda + \mu)(\bar{P}(z) - P_0) = \mu(\bar{P}(z) - P_0)\bar{B}(c) + \lambda\bar{A}(z)\bar{P}(z). \quad (1.8)$$

Подставляя в (1.8) выражения для производящей функции $\bar{P}(z) = \frac{P_0 + (1 - P_0 - c) \cdot z}{1 - c \cdot z}$, получим

$$\bar{A}(z) = \frac{(1 - c)z}{P_0 + (1 - P_0 - c)z}. \quad (1.9)$$

Откуда следует, что $a(n)$ распределено по геометрическому закону

$$a(n) = (1 - a)a^{n-1}, \quad \text{где } a = \frac{P_0 + c - 1}{P_0}. \quad (1.10)$$

Из проведенных рассуждений вытекает

Теорема 1. Пусть выполнено (1.2). Для того чтобы стационарное распределение $X(t)$ имело форму смещенного геометрического распределения (1.5), необходимо и достаточно, чтобы $P_0 < \frac{1}{1+\rho}$ и существовала абсолютно монотонная функция $\check{A}(z)$, удовлетворяющая $\check{A}(1) = 1$ и равенству (1.9). Тогда $\check{A}(z)$ будет задавать производящую функцию размеров поступающих групп.

2. Модель сети с групповыми перемещениями. Рассмотрим открытую сеть массового обслуживания, состоящую из двух узлов того же типа что и система, рассмотренная в разделе 1. Обслуженная в i -м узле группа заявок мгновенно покидает сеть, посылая сообщение в j -й узел с вероятностью π_{ij} , а с вероятностью π_{i0} не посылая никаких сообщений ($i, j = 1, 2$ $\sum_{j=0}^2 \pi_{ij} = 1$, $p_{ii} = 0$).

Предположим, что расширенная матрица маршрутов неприводима. Тогда система линейных уравнений трафика

$$\begin{aligned} \tilde{\gamma}_1 &= \lambda_1 + \gamma_2 \pi_{2,1} \\ \tilde{\gamma}_2 &= \lambda_2 + \gamma_1 \pi_{1,2} \end{aligned} \tag{2.1}$$

имеет единственное строго положительное решение. Где γ_i – интенсивность потока сообщений, выходящих из i -го узла, а $\tilde{\gamma}_i$ – интенсивность потока сообщений, поступающих на i -й узел.

Предположим, что выполнено условие эргодичности

$$\gamma_i m_{A_i} < \mu_i m_{B_i}, \quad i = 1, 2. \tag{2.2}$$

Покажем, что для сети, состоящей из двух узлов, стационарное распределение будет иметь форму

$$P(n) = P_1(n_1)P_2(n_2), \tag{2.3}$$

где $\{P_i(n_i)\}$ – вычисляются по формуле $P_i(n_i) = (1 - P_{i,0})(1 - c_i)c_i^{n_i}$.

Запишем уравнения равновесия для двух узлов сети

$$P(0,0)[\lambda_1 + \lambda_2] = \sum_{k=1}^{\infty} P(k,0)\bar{B}_1(k)\mu_1\pi_{10} + \sum_{k=1}^{\infty} P(0,k)\bar{B}_2(k)\mu_2\pi_{20}, \tag{2.4}$$

$$P(n_1,0)[\lambda_1 + \lambda_2 + \mu_1] = \sum_{k=1}^{n_1} P(n_1 - k,0)\lambda_1 a_1(k) + \sum_{k=1}^{\infty} P(n_1 + k,0)b_1(k)\mu_1\pi_{10} + \tag{2.5}$$

$$+ \sum_{k=1}^{\infty} P(n_1, k)\bar{B}_2(k)\mu_2\pi_{20} + \sum_{k=1}^{\infty} \sum_{l=1}^{n_1} P(n_1 - l, k)\mu_2\bar{B}_2(k)\pi_{21}a_1(l),$$

$$P(0, n_2)[\lambda_1 + \lambda_2 + \mu_2] = \sum_{k=1}^{n_2} P(0, n_2 - k)\lambda_2 a_2(k) + \sum_{k=1}^{\infty} P(0, n_2 + k)b_2(k)\mu_2\pi_{20} + \tag{2.6}$$

$$+ \sum_{k=1}^{\infty} P(k, n_2)\bar{B}_1(k)\mu_1\pi_{10} + \sum_{k=1}^{\infty} \sum_{l=1}^{n_2} P(k, n_2 - l)\mu_1\bar{B}_1(k)\pi_{12}a_2(l),$$

$$P(n_1, n_2)[\lambda_1 + \lambda_2 + \mu_1 + \mu_2] = \sum_{k=1}^{\infty} P(n_1 - k, n_2)\lambda_1 a_1(k) + \sum_{k=1}^{n_2} P(n_1, n_2 - k)\lambda_2 a_2(k) + \tag{2.7}$$

$$+ \sum_{k=1}^{\infty} P(n_1 + k, n_2)b_1(k)\mu_1\pi_{10} + \sum_{k=1}^{\infty} P(n_1, n_2 + k)b_2(k)\mu_2\pi_{20} +$$

$$+ \sum_{k=1}^{\infty} \sum_{l=1}^{n_2} P(n_1 + k, n_2 - l)\mu_1 b_1(k)\pi_{12}a_2(l) + \sum_{k=1}^{\infty} \sum_{l=1}^{n_1} P(n_1 - l, n_2 + k)\mu_2 b_2(k)\pi_{21}a_1(l).$$

С учетом (2.3), (2.4) примет вид

$$\lambda_1 + \lambda_2 = \pi_{10}\tilde{\gamma}_1 + \pi_{20}\tilde{\gamma}_2, \quad (2.8)$$

а при подстановке (2.3) в (2.5), (2.6), (2.7) равенства превратятся в тождества.

Из всего вышеизложенного вытекает

Теорема 2. При выполнении условий (2.2) процесс $n(t)$ эргодичен. Для того, что бы его стационарное распределение представлялось в обобщенной форме произведения (2.3), необходимо выполнения равенства (2.8) и условий теоремы 1 для каждого из узлов.

Литература

1. Miyazawa M., Taylor P.G. A Geometric Product-form Distribution for a Queueing Network with Non-standard Batch Arrivals and Batch Transfers // Adv.Appl.Prob. 1997. V.29. No. 2. P.1-22.
2. Chao X. and Pinedo M. On Generalized Networks of Queues with Positive and Negative Arrivals // Prob. Eng. Inf. Sci. 1993. V.7. P.301-304.
3. Chao X., Pinedo M. and Shaw D. A Network of Assembly Queues with Product-form Solution // J. Appl. Prob. 1996. V.33. P.858-869.

УДК 519.6

МЕТОД СЕКУЩИХ ДЛЯ МИНИМИЗАЦИИ ФУНКЦИЙ ОДНОЙ ПЕРЕМЕННОЙ

Костюк А.Ю.

УО «Брестский государственный технический университет», г. Брест
Научный руководитель В.М.Ракецкий, к.ф.-м.н, доцент

Введение. Методы минимизации функций одной переменной занимают важное место в теории и практике численных методов оптимизации, поскольку они часто используются при разработке методов минимизации функций n переменных. Наиболее популярными из них являются прямые методы (или нулевого порядка), не требующие вычисления производных.

Главное требование, которое предъявляется к прямым методам, можно сформулировать так: достичь заданную точность отыскания точки экстремума при минимальном количестве вычислений функции. Именно эта идея лежит в основе широко известных методов Фибоначчи, “золотого” сечения и квадратичной интерполяции [1,2]. Методы Фибоначчи и “золотого” сечения не предполагают и не требуют от функции никаких дополнительных свойств, кроме свойства унимодальности на заданном отрезке. Они одинаково хороши как для дифференцируемых, так и не дифференцируемых функций. Метод квадратичной интерполяции разработан из предположения, что минимизируемая функция является гладкой, хотя бы один раз дифференцируемой, однако его применение во многих случаях дает хорошие результаты и для недифференцируемых функций.

Ниже приводится описание прямого метода минимизации функции одной переменной, который получил условное название метода секущих, и результаты численного эксперимента по его сравнению с другими аналогичными методами.

Идея метода. Если функция $f(x)$ унимодальна на отрезке $[a, b]$, то, зная внутри этого отрезка две точки x_1, x_2 и значения функции в них, можно уменьшить отрезок локализации экстремума, отбросив его часть, заведомо не содержащую точку минимума:

$[a, x_1]$, если $f(x_1) < f(x_2)$, или $[x_2, b]$, если $f(x_1) > f(x_2)$. Этот принцип так или иначе используется во всех методах минимизации унимодальных функций. Однако кроме значений функции в точках x_1, x_2 обычно известны и значения функции на концах отрезка a, b . С помощью информации о четырех точках можно получить новую информацию. Для этого соединим (см. рис. 1) точки $(a, f(a)), (x_1, f(x_1))$ и $(x_1, f(x_1)), (b, f(b))$ прямыми.

Левая прямая есть линейная аппроксимация левой ветви функции, правая – правой, а абсцисса точки пересечения прямых – приближение к точке минимума. Если ситуация сложится именно так, как показано на рисунке, то после нахождения пятой точки можно от отрезка $[a, b]$ сразу перейти к отрезку $[x_1, x_2]$. Кроме этого можно контролировать процесс минимизации по двум последовательно найденным приближениям: если x^{*1} и x^{*2} – два последовательно найденных приближения к точке минимума, то x^{*2} принимаем в качестве точки экстремума, если $|x^{*1} - x^{*2}| \leq \epsilon$, где ϵ – заданная точность отыскания экстремума.

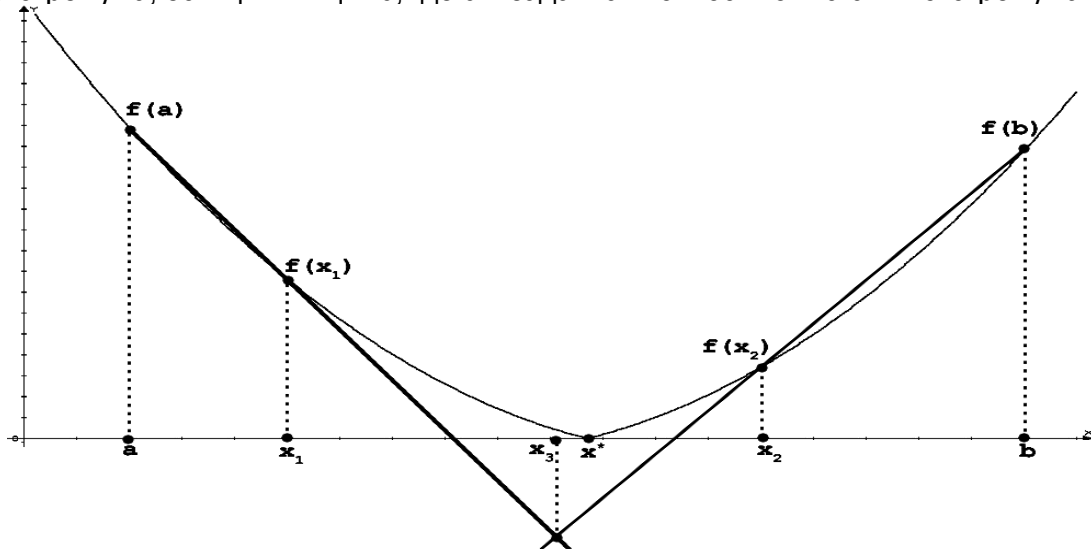


Рисунок 1 – Графическая интерпретация метода секущих

Никаких предположений о поведении функции $f(x)$ на отрезке $[a, b]$, кроме унимодальности, не высказывалось. Поэтому этот метод приемлем для поиска минимума любых унимодальных функций. Однако характер аппроксимации подсказывает, что метод должен давать хорошие результаты для негладких функций и особенно для функций, которые близки по своему поведению к линейным негладким функциям типа $|x - x^*|$.

Описание метода. Обсудим сначала, как располагать точки x_1, x_2 внутри исходного отрезка $[a, b]$. Из последующего описания будет видно, что какого-то регулярного, заранее predeterminedного размещения точек внутри отрезка локализации экстремума, как это есть в методах Фибоначчи или “золотого” сечения, сконструировать не удастся. Поэтому первоначальное размещение точек в принципе может быть любым, лишь бы точки x_1, x_2 не совпадали друг с другом. Предлагаются два способа:

- 1) точки x_1, x_2 располагаются равномерно на отрезке $[a, b]$ и делят его на 3 равные части;
- 2) сначала в середине отрезка $[a, b]$ располагается точка x_1 . Точка x_2 помещается в середину одного из двух получившихся отрезков в зависимости от соотношения значений функции на концах отрезка $[a, b]$. Если $f(a) < f(b)$, то x_2 – середина отрезка $[a, x_1]$; при $f(a) > f(b)$ точка x_2 – середина отрезка $[x_1, b]$.

Оба способа допускают дополнительный анализ: после построения точки x_1 можно проанализировать соотношение между значениями $f(a), f(x_1)$ и $f(b)$. Если $f(a) < f(x_1) < f(b)$, то в силу унимодальности функции точка экстремума $x^* \in [a, x_1]$. В случае $f(a) > f(x_1) > f(b)$ точка минимума $x^* \in [x_1, b]$. В этих случаях можно сразу, без построения точки x_2 , уменьшить отрезок локализации экстремума.

Рассмотрим теперь общую конструкцию метода. Как следует из его идеи, для метода секущих типичной является схема, когда от отрезка $[a, b]$ происходит переход к отрезку $[a^1, b^1] = [x_1, x_2]$. Внутри последнего остается точка, в которой пересекаются секущие (для определенности, обозначим ее x_1^1). Эту точку, по аналогии с другими методами, необходимо использовать в дальнейших вычислениях. Возникает вопрос: как в этом случае выбрать вторую точку x_2^1 внутри отрезка $[a^1, b^1]$? Один из вариантов – симметрично точке x_1^1 относительно середины отрезка $[a_1, b_1]$. Однако расположение точки x_1^1 внутри отрезка $[a^1, b^1]$ не предсказуемо и при симметричном расположении точки x_2^1 нельзя гарантировать, что следующий отрезок $[a^2, b^2]$ будет существенно короче отрезка $[a^1, b^1]$. Поэтому точку x_2^1 предлагается располагать в середине большего из отрезков $[a_1, x_2^1]$, $[x_2^1, b^1]$. В этом случае следующий отрезок будет более чем в два раза короче предыдущего (если переход от $[a^1, b^1]$ к $[a^2, b^2]$ произойдет по типичной схеме).

Рассуждения предыдущего абзаца основывались на типичной схеме. Но она не является единственной. В процессе вычислений возможна нетипичная схема №1, когда точка пересечения секущих не принадлежит отрезку $[x_1, x_2]$. В этом случае, для упрощения анализа, перенумеруем три внутренних точки таким образом, чтобы выполнялось соотношение $x_1 < x_2 < x_3$. После этого новый отрезок строится по одному из 3-х правил:

- 1) $[a^1, b^1] = [x_1, x_3]$, если $f(x_2) < \min\{f(x_1), f(x_3)\}$;
- 2) $[a^1, b^1] = [a, x_1]$, если $f(x_1) < f(x_2) < f(x_3)$;
- 3) $[a^1, b^1] = [x_3, b]$, если $f(x_1) > f(x_2) > f(x_3)$.

Наконец, возможна нетипичная схема №2, когда точка пересечения секущих совпадает в пределах заданной точности с одной из точек x_1, x_2 или вообще не принадлежит отрезку $[a, b]$. В этом случае новый отрезок строится классическим способом (для упрощения считаем, что $x_1 < x_2$):

- 1) $[a^1, b^1] = [a, x_1]$, если $f(x_1) < f(x_2)$;
- 2) $[a^1, b^1] = [x_2, b]$, если $f(x_1) > f(x_2)$.

С целью упрощения при анализе описанных выше ситуаций сознательно не рассматривались соотношения \leq, \geq . Для описания общей схемы метода они не имеют принципиального значения.

В заключение описания метода приведем формулы для вычисления абсциссы точки пересечения секущих (обозначим ее как x_s):

$$x_s = \frac{B_2(x_2, b) - B_1(a, x_1)}{A_1(a, x_1) - A_2(x_2, b)},$$

$$\text{где } A_1(a, x_1) = \frac{f(a) - f(x_1)}{a - x_1}; \quad A_2(x_2, b) = \frac{f(x_2) - f(b)}{x_2 - b}; \quad B_1(a, x_1) = \frac{-x_1 f(a) + a f(x_1)}{a - x_1};$$

$$B_2(x_2, b) = \frac{-b f(x_2) + x_2 f(b)}{x_2 - b}.$$

Численный эксперимент. Для проверки эффективности метода секущих был проведен численный эксперимент, в котором метод секущих сравнивался по количеству итераций и вычислений функции с методами “золотого” сечения и квадратичной интерполяции. При этом реализован был алгоритм, в котором точки x_1, x_2 для начального отрезка строились по правилу 2 с дополнительным анализом значения в точке x_1 .

Негладкие функции:

Функции	[a; b]	Количество вычислений функции		
		“Золотое сечение”	Метод секущих	Квадратичной интерполяции
1. $F(x) = x - \pi $	[3;4]	20	8	14
	[0;4]	22	8	14
2. $F(x) = x^2 - \pi^2 $	[3;4]	20	12	16
	[0;4]	22	19	19
3. $F(x) = \begin{cases} \pi^2 - x^2, & \text{если } x \leq \pi \\ x - \pi, & \text{если } x > \pi \end{cases}$	[3;4]	20	10	28
	[0;4]	22	12	30
4. $F(x) = x - \pi + 0.1(x - \pi)^2$	[3;4]	20	12	13
	[0;4]	22	16	17

Гладкие функции:

Функции	[a; b]	Количество вычислений функции		
		“Золотое сечение”	Метод секущих	Квадратичной интерполяции
1. $F(x) = x + \frac{1}{x}$	[0.5;7]	24	33	22
	[0.9;6]	23	33	13
2. $F(x) = \frac{x - 5}{4x^2 - 25x + 40}$	[1;4]	22	29	11
	[3;6]	22	28	11

Численный эксперимент подтверждает предположение о том, что для негладких функций метод секущих будет более эффективен, чем для гладких. Особенно он эффективен для функций, близких к линейным негладким функциям.

Литература

1. Габасов, Р. Методы оптимизации / Р. Габасов, Ф.М. Кириллов – Минск: Изд-во БГУ, 1981.
2. Химмельблау, Д. Прикладное нелинейное программирование / Д. Химмельблау – М: Мир, 1975.

УДК 519.24

**ИССЛЕДОВАНИЕ МАТЕМАТИЧЕСКОГО ОЖИДАНИЯ И ДИСПЕРСИИ
ОСРЕДНЕННОЙ ОЦЕНКИ ВЗАИМНОЙ СПЕКТРАЛЬНОЙ ПЛОТНОСТИ**

Кулак Т.Н.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель - Мирская Е.И., кандидат физ.-мат. наук, доцент

Рассмотрим r -мерный стационарный случайный процесс $X(t) = \{X_a(t), a = \overline{1, r}\}$, $t \in Z$, $a = \overline{1, r}$, с $MX(t) = 0$, неизвестной взаимной спектральной плотностью $f_{ab}(\lambda)$, $\lambda \in \Pi = [-\pi, \pi]$, $a, b = \overline{1, r}$.

$$f_{ab}(\lambda) = \frac{1}{2\pi} \sum_{t=-\infty}^{\infty} R_{ab}(t) e^{-i\lambda t} \tag{1}$$

Пусть $X_a(0), X_a(1), \dots, X_a(T-1)$ - T последовательных наблюдений за составляющей $X_a(t)$ процесса $X(t)$, $t \in Z$, $a = \overline{1, r}$ и $T = LN - (L-1)K$, где L - число пересекающихся интервалов, содержащих по N наблюдений, а K принимает целочисленные значения, $0 \leq K \leq N$.

Используя метод Уэлча [1], в качестве оценки взаимной спектральной плотности процесса исследована статистика вида

$$\hat{f}_{ab}^{(T)}(\lambda) = \frac{1}{L} \sum_{l=1}^L I_{ab}(\lambda, l), \quad (2)$$

где

$$I_{ab}(\lambda, l) = \frac{1}{2\pi \sum_{p=0}^{N-1} h_a^N(p) h_b^N(p)} H_a(\lambda, l) \overline{H_b(\lambda, l)}, \quad (3)$$

где $l = \overline{1, L}$, $a, b = \overline{1, r}$, $\lambda \in \Pi$, модифицированная периодограмма на l -ом интервале разбиения, а $H_a(\lambda, l)$ задано выражением

$$H_a(\lambda, l) = \sum_{t=0}^{N-1} h_a^N(t) X_a(t + (l-1)(N-K)) e^{-i\lambda(t+(l-1)(N-K))}, \quad (4)$$

$l = \overline{1, L}$, $a, b = \overline{1, r}$, $\lambda \in \Pi$, причем наблюдения сглаживаются одним и тем же окном просмотра данных $h_a^N(t)$, $t \in Z$.

Исследованы некоторые статические свойства оценки $\hat{f}_{ab}^{(T)}(\lambda)$, $\lambda \in \Pi$, $a, b = \overline{1, r}$. Найдены математическое ожидание и дисперсия построенной оценки.

Теорема 1. Математическое ожидание оценки взаимной спектральной плотности $\hat{f}_{ab}^{(T)}(\lambda)$, $\lambda \in \Pi$, $a, b = \overline{1, r}$, задаваемой соотношением (2), имеет вид

$$M \hat{f}_{ab}^{(T)}(\lambda) = \int_{\Pi} f_{ab}(x + \lambda) \Phi_{ab}(x) dx, \quad (5)$$

где

$$\Phi_{ab}(x) = \left[2\pi \sum_{t=0}^{N-1} h_a^N(t) h_b^N(t) \right]^{-1} \varphi_a(x) \overline{\varphi_b(x)}, \quad (6)$$

$$\varphi_a(x) = \sum_{t=0}^{N-1} h_a^N(t) e^{itx}, \quad (7)$$

$x \in \Pi$, $a, b = \overline{1, r}$.

Доказательство. Подставляя вместо $\hat{f}_{ab}^{(T)}(\lambda)$ ее выражение в явном виде, используя соотношения (4), (3) и учитывая свойства математического ожидания, получим

$$\begin{aligned} M \hat{f}_{ab}^{(T)}(\lambda) &= \frac{1}{L} \sum_{l=1}^L M I_{ab}(\lambda, l) = \frac{1}{L} \sum_{l=1}^L \frac{1}{2\pi \sum_{p=0}^{N-1} h_a^N(p) h_b^N(p)} M \{H_a(\lambda, l) \overline{H_b(\lambda, l)}\} = \\ &= \frac{1}{L} \sum_{l=1}^L \frac{1}{2\pi \sum_{p=0}^{N-1} h_a^N(p) h_b^N(p)} M \left[\sum_{t=0}^{N-1} h_a^N(t_1) X_a(t_1 + (l-1)(N-K)) e^{-i\lambda(t_1+(l-1)(N-K))} \right] \times \\ &\times \sum_{t=0}^{N-1} h_b^N(t_2) X_b(t_2 + (l-1)(N-K)) e^{-i\lambda(t_2+(l-1)(N-K))} = \frac{1}{L} \sum_{l=1}^L \frac{1}{2\pi \sum_{p=0}^{N-1} h_a^N(p) h_b^N(p)} \times \\ &\times \sum_{t_1=0}^{N-1} \sum_{t_2=0}^{N-1} h_a^N(t_1) h_b^N(t_2) e^{-i\lambda(t_1-t_2)} R_{ab}(t_1 - t_2). \end{aligned}$$

Учитывая связь взаимной ковариационной функции и взаимной спектральной плотности

$$R_{ab}(t) = \int_{\Pi} f_{ab}(x)e^{itx} dx,$$

получим требуемый результат. Теорема доказана.

Доказано, что если взаимная спектральная плотность $f_{ab}(\lambda)$, $a, b = \overline{1, r}$, заданная соотношением (1), непрерывна в точках λ_1, λ_2 и ограничена на Π , семиинвариантная спектральная плотность 4-го порядка ограничена на Π^3 , окна просмотра данных ограничены единицей, то

$$\lim_{N \rightarrow \infty} D \hat{f}_{ab}^{(T)}(\lambda) = \begin{cases} \frac{1}{L} C_1 f_{aa}(\lambda) f_{bb}(\lambda), & \text{при } \lambda \neq 0 \pmod{\pi}, \\ \frac{1}{L} (f_{ab}(0) f_{ba}(0) + C_1 f_{aa}(0) f_{bb}(0)), & \text{при } \lambda = 0 \pmod{\pi}, \end{cases} \quad (8)$$

где C_1 - задано соотношением

$$C_1 = \frac{\int_0^1 [h_a^N(x)]^2 dx \int_0^1 [h_b^N(x)]^2 dx}{\left(\int_0^1 h_a^N(x) h_b^N(x) dx\right)^2},$$

$\lambda \in \Pi$, $a, b = \overline{1, r}$.

Теорема 2. Если взаимная спектральная плотность $f_{ab}(x)$ непрерывна в точке $x = \lambda$ и ограничена на Π , окна просмотра данных $h_a^N(t)$, $a = \overline{1, r}$, $t \in R$, ограничены единицей и имеют ограниченную постоянную вариацию, то для оценки $\hat{f}_{ab}^{(T)}(\lambda)$, $\lambda \in \Pi$, $a, b = \overline{1, r}$, заданной выражением (2), справедливо соотношение

$$\lim_{N \rightarrow \infty} M \hat{f}_{ab}^{(T)}(\lambda) = f_{ab}(\lambda).$$

Построен график оценки спектральной плотности (2), полученной на основании данных о солнечной активности по Вольфу в каждом полугодии с 1749 г. по 1901 г. для окна Фейера.

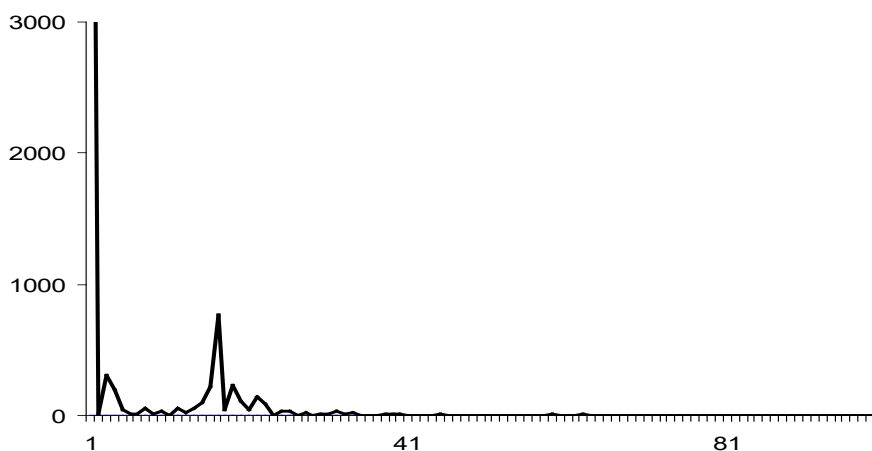


Рисунок 1 – График оценки спектральной плотности (2) для окна Фейера, полученной на основании данных о солнечной активности по Вольфу в каждом полугодии с 1749 г. по 1901 г.

Литература

1. Welch, P. D. The use of FFT for the estimation of power spectra / P. D. Welch // IEEE Trans. Electroacoust. – 1967. – Vol. 15., № 2. – P. 70–73.

2. Бриллинджер, Д. Временные ряды. Обработка данных и теория / Д. Бриллинджер. - М.: Мир, 1980. – 536 с.

УДК 519.95

ИНТЕГРАЛ ДЮАМЕЛЯ И ЕГО ПРИЛОЖЕНИЯ**Лапич С.В.**

УО «Брестский государственный технический университет», г. Брест

Научный руководитель: доцент Тузик Т.А.

При решении прикладных задач, в частности задачи анализа входных процессов линейных динамических систем (линейны электрических цепей), возникает необходимость в решении задачи, которая по своей сути сводится к решению ЛДУ.

Требуется решить ЛДУ следующего вида:

$$x^{(n)} + a_1 x^{(n-1)} + a_2 x^{(n-2)} + \dots + a_n x = f(t) \quad (1)$$

При нулевых начальных условиях $x(0) = x'(0) = x''(0) = \dots = x^{(n-1)}(0) = 0$

Решаем уравнение (1), когда $f(t)=1$, и получаем уравнение следующего вида:

$$x^{(n)} + a_1 x^{(n-1)} + a_2 x^{(n-2)} + \dots + a_n x = 1 \quad (2)$$

при тех же начальных условиях. После преобразования по Лапласу получаем: $A(p)X(p) = F(p)$, для уравнения (1) и $A(p)X_1(p) = \frac{1}{p}$ – для уравнения (2). Преобразовывая данные выражения, получим решение ДУ через интеграл Дюамеля:

$$X(p) = \frac{F(p)}{A(p)}, \quad X_1(p) = \frac{1}{pA(p)}, \quad X(p) = pX_1(p)F(p) = \int_0^t f(\tau)x_1'(t-\tau)d\tau;$$

$$x(t) = L^{-1}X(p) = L^{-1}(pX_1(p)F(p)) = \int_0^t f(\tau)x_1'(t-\tau)d\tau \quad (3)$$

Интеграл Дюамеля позволяет находить решения ЛДУ при нулевых начальных условиях, не находя изображения правой части уравнения (1). Формулу целесообразно использовать, когда производится исследование какой-либо динамической системы, работа которой описывается левой частью уравнения (1). Для неё один раз находим решение уравнения (2) с начальными нулевыми условиями, а затем ищутся решения уравнения (1) при любых $f(t)$.

Рассмотрим два примера использования формулы Дюамеля для решения линейных ДУ второго порядка с нулевыми начальными условиями.

Пример 1.

Условие: $x'' - x' = \frac{e^{2t}}{2 + e^t}$, $x(0) = x'(0) = 0$.

Решаем ДУ, когда $f(t)=1$ и начальные условия нулевые:

$$x'' - x' = 1, \quad x_1(t) = e^t - t - 1, \quad x_1'(t) = e^t - 1.$$

Тогда решение заданного ДУ получим по формуле (3).

$$x(t) = \int_0^t \frac{e^{2\tau}}{2 + e^\tau} (e^{t-\tau} - 1) d\tau;$$

$$\begin{aligned} \int_0^t \frac{e^{2\tau}}{2 + e^\tau} (e^{t-\tau} - 1) d\tau &= \int_0^t \frac{e^{\tau+t}}{2 + e^\tau} d\tau - \int_0^t \frac{e^{2\tau}}{e^\tau + 2} d\tau = e^t \int_0^t \frac{e^\tau}{2 + e^\tau} d\tau - \int_0^t \frac{e^\tau}{e^\tau + 2} de^\tau = e^t \cdot \ln(e^\tau + 2) \Big|_0^t - \\ &- (e^\tau) \Big|_0^t + 2 \ln(e^\tau + 2) \Big|_0^t = e^t (\ln(e^t + 2) - \ln 3) - (e^t - 1) + 2 \ln(e^t + 2) \cdot \ln 3 = e^t \ln \frac{e^t + 2}{3} - e^t + 1 + \\ &+ 2 \ln \frac{e^t + 2}{3}; \end{aligned}$$

$$x(t) = (e^t + 2) \ln \frac{e^t + 2}{3} - e^t + 1.$$

Аналогичным образом решаем задачу Коши для ДУ вида:

$$x'' + x' = \frac{1}{2 + \cos t}, \quad x(0) = x'(0) = 0.$$

Решение. $x'' + x' = 1$, $p^2 X(p) + X(p) = \frac{1}{p}$, $X(p) = \frac{1}{p} \cdot \frac{1}{p^2 + 1}$;

$$X(p) = \frac{-p}{p^2 + 1} + \frac{1}{p}, \quad x_1(t) = 1 - \cos t, \quad x_1' = \sin t;$$

$$x(t) = \int_0^t \frac{1}{2 + \cos \tau} \sin(t - \tau) d\tau;$$

$$\begin{aligned} \int_0^t \frac{1}{2 + \cos \tau} \sin(t - \tau) d\tau &= \int_0^t \frac{\sin t \cos \tau - \cos t \sin \tau}{2 + \cos \tau} d\tau = \sin t \int_0^t \frac{\cos \tau + 2 - 2}{\cos \tau + 2} d\tau - \cos t \int_0^t \frac{\sin \tau}{2 + \cos \tau} d\tau = \\ &= t \sin t + 2 \sin t \int_0^t \frac{d\tau}{3 \cos^2 \frac{\tau}{2} + \sin^2 \frac{\tau}{2}} + \cos t \cdot \ln(2 + \cos \tau) \Big|_0^t; \end{aligned}$$

После интегрирования получаем обычное решение исходной задачи.

$$x(t) = t \sin t - \frac{4}{\sqrt{3}} \operatorname{arctg} \frac{\operatorname{tg} \frac{t}{2}}{\sqrt{3}} + \cos t \cdot \frac{\ln(2 + \cos t)}{3}.$$

Расчет переходных процессов с использованием интеграла Дюамеля

Зная реакцию цепи на единичное возмущающее воздействие, т.е. функцию переходной проводимости $g(t)$ или (и) переходную функцию по напряжению $h(t)$, можно найти реакцию цепи на воздействие произвольной формы. В основе метода – метода расчета с помощью интеграла Дюамеля – лежит принцип наложения.

При использовании интеграла Дюамеля для разделения переменной, по которой производится интегрирование, и переменной, определяющей момент времени, в который определяется ток в цепи, первую принято обозначать как τ , а вторую – как t .

Пусть в момент времени $t=0$ к цепи с нулевыми начальными условиями (пассивному двухполюснику ПД на рис. 1) подключается источник с напряжением $u(\tau)$ произвольной формы. Для нахождения тока i в цепи заменим исходную кривую ступенчатой (см. рис. 2), после чего с учетом, что цепь линейна, просуммируем токи от начального скачка напряжения $u(0)$ и всех ступенек напряжения до момента t , вступающих в действие с запаздыванием по времени.

В момент времени t составляющая общего тока, определяемая начальным скачком напряжения $u(0)$, равна $u(0)g(t)$.

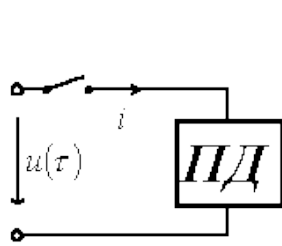


Рисунок 1

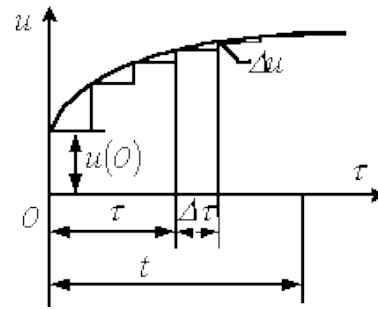


Рисунок 2

В момент времени $\tau + \Delta\tau$ имеет место скачок напряжения $\Delta u \approx u'(\tau)\Delta\tau$, который с учетом временного интервала от начала скачка до интересующего момента времени t обусловит составляющую тока $u'(\tau)g(t-\tau-\Delta\tau)\Delta\tau$.

Полный ток $i(t)$ в момент времени t равен, очевидно, сумме всех составляющих тока от отдельных скачков напряжения с учетом $u(0)g(t)$, т.е.

$$i(t) = u(0)g(t) + \sum u'(\tau)g(t-\tau-\Delta\tau)\Delta\tau$$

Переходя к пределу, когда $\Delta\tau$ стремиться к 0, получим, что полный ток, рассчитываемый с помощью интеграла Дюамеля, равен:

$$i(t) = u(0)g(t) + \int_0^t u'(\tau)g(t-\tau)d\tau \quad (4)$$

Следует отметить, что с использованием интеграла Дюамеля можно определять также напряжение. При этом в формуле (4) вместо переходной проводимости $g(t)$ будет входить переходная функция по напряжению.

Пример. Пусть известна переходная проводимость $g(t)$, равная

$\frac{1}{R}(1 - e^{-\frac{R}{L}t})$, при напряжении цепи $u(t) = 1000e^{-5t}$ В, сопротивлении $R = 10$ Ом и индукции $L = 1$ Гн. Рассчитаем значение тока в любой момент времени t .

Решение.

В данном случае:

1. $g(t) = 0,1(1 - e^{-10t})$;
2. $g(t-\tau) = 0,1(1 - e^{-10(t-\tau)})$;
3. $u'(\tau) = -5000e^{-5\tau}$;

$$4. \quad i(t) = u(0)g(t) + \int_0^t u'(\tau)g(t-\tau)d\tau$$

$$i(t) = 100(1 - e^{-10t}) - 500 \int_0^t e^{-5\tau} d\tau + 500 e^{-10t} \int_0^t e^{5\tau} d\tau;$$

После интегрирования получаем решение исходной задачи:

$$i(t) = 200(e^{-5t} - e^{-10t}) \text{ А.}$$

Литература

1. Краснов, М.Л. Функции комплексного переменного. Операционное исчисление. Теория устойчивости / М.Л. Краснов, А.И. Киселёв, Г.И. Макаренко – М.: Наука, 1981.
2. Пантелеев, А.В. Теория функций комплексного переменного и операционное исчисление в примерах и задачах: учебное пособие / А.В. Пантелеев, А.С. Якимова – М.: Высш. шк., 2001.

УДК 519.6+517.983.54

РЕШЕНИЕ ОПЕРАТОРНЫХ УРАВНЕНИЙ ДВУХШАГОВЫМ ИТЕРАЦИОННЫМ МЕТОДОМ

Лялюк Ю.В.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Матысик О.В., канд. физ.-мат. наук, доцент*

Поскольку некорректные задачи постоянно возникают в многочисленных приложениях математики, то проблема их решения и разработка методов их решения является актуальной. В работе предлагается двухшаговый метод итерации решения некорректных задач. Цель исследования – доказать сходимость метода и получить оценки погрешности в исходной норме гильбертова пространства. Методологической основой исследования является общая теория некорректных задач, элементы функционального и математического анализа, вычислительной математики.

В гильбертовом пространстве H решается уравнение первого рода

$$Ax = y, \tag{1}$$

где A – ограниченный положительный и самосопряжённый оператор, в предположении, что нуль принадлежит спектру оператора A , но не является его собственным значением. Тогда задача о разрешимости уравнения (1) является некорректной. Если решение уравнения (1) существует, то будем искать его с помощью явного двухшагового итерационного метода

$$x_n = 2(E - \alpha A)x_{n-1} - (E - \alpha A)^2 x_{n-2} + \alpha^2 Ay, x_0 = x_1 = 0. \tag{2}$$

Обычно правая часть уравнения (1) известна с некоторой точностью δ , т.е. известно y_δ такое, что $\|y - y_\delta\| \leq \delta$. Тогда метод (2) примет вид

$$x_{n,\delta} = 2(E - \alpha A)x_{n-1,\delta} - (E - \alpha A)^2 x_{n-2,\delta} + \alpha^2 Ay_\delta, x_{0,\delta} = x_{1,\delta} = 0. \tag{3}$$

Для методов (2) и (3) справедливы теоремы.

Теорема 1. *Итерационный метод (2) при условии $0 < \alpha < \frac{2}{\|A\|}$ сходится в исходной норме гильбертова пространства.*

Теорема 2. *Метод (3) сходится в исходной норме гильбертова пространства при условии $0 < \alpha \leq \frac{5}{4\|A\|}$, если число итераций n выбирать в зависимости от δ так, чтобы $n\delta \rightarrow 0, n \rightarrow \infty, \delta \rightarrow 0$.*

Теорема 3. *При условии $0 < \alpha \leq \frac{5}{4\|A\|}$ и $x = A^s z, s > 0$ для итерационного двухшагового метода (3) справедлива оценка погрешности*

$$\|x - x_{n,\delta}\| \leq s^s (s + 2) [(n - 1)\alpha e]^{-s} \|z\| + \left(\frac{5}{4}\right) (n - 1)\delta\alpha.$$

Теорема 4. *Оптимальная по n оценка погрешности для метода (3) имеет вид*

$$\|x - x_{n,\delta}\|_{\text{opt}} \leq \left(\frac{5}{4}\right)^{s/(s+1)} \delta^{s/(s+1)} e^{-s/(s+1)} (s + 1)(s + 2)^{1/(s+1)} \|z\|^{1/(s+1)} \tag{4}$$

и получается при

$$n_{\text{opt}} = 1 + \left(\frac{5}{4}\delta\right)^{-1/(s+1)} e^{-s/(s+1)} s(s+2)^{1/(s+1)} \|z\|^{1/(s+1)} \alpha^{-1}. \quad (5)$$

Замечание 1. Оптимальная по n оценка погрешности (4) имеет порядок $O\left(\frac{s}{\delta^{s+1}}\right)$,

и он оптимален в классе задач с истокообразно представимыми решениями $x = A^s z, s > 0$ [1].

Замечание 2. Оптимальная оценка погрешности не зависит от α , но от итерационного параметра α зависит n_{opt} (5), и, значит, объём вычислительной работы.

Для его уменьшения следует выбирать α из условия $0 < \alpha \leq \frac{5}{4\|A\|}$ и так, чтобы

n_{opt} было целым.

Предложенный метод может быть успешно применён для решения некорректных задач, встречающихся в спектроскопии, акустике, синтезе антенн, математической обработке данных эксперимента, теплопроводности.

Литература

1. Вайникко, Г. М. Итерационные процедуры в некорректных задачах / Г. М. Вайникко, А. Ю. Веретенников. – М. Наука, 1986. – 178 с.

УДК 519.24

ИССЛЕДОВАНИЕ МАТЕМАТИЧЕСКОГО ОЖИДАНИЯ СОСТОЯТЕЛЬНОЙ ОЦЕНКИ ВЗАИМНОЙ СПЕКТРАЛЬНОЙ ПЛОТНОСТИ

Марчук А.Ю.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель - Мирская Е.И., кандидат физ.-мат. наук, доцент

Рассмотрим действительный стационарный случайный процесс $X(t) = \{X_a(t), a = \overline{1, r}\}$, $t \in Z$, с $MX(t) = 0$, неизвестной взаимной спектральной плотностью $f_{ab}(\lambda)$, $\lambda \in \Pi = [-\pi, \pi]$, $a, b = \overline{1, r}$.

Пусть $X_a(0), X_a(1), \dots, X_a(T-1)$ - T последовательных наблюдений, полученных через равные промежутки времени, за составляющей $X_a(t)$ процесса $X(t)$, $t \in Z$, $a = \overline{1, r}$.

Предполагаем, что число наблюдений T представимо в виде $T = LN - (L-1)K$, где L - число пересекающихся интервалов, содержащих по N наблюдений, а K принимает целочисленные значения, $0 \leq K \leq N$.

Используя методику Бриллинджера Д. [2], в качестве оценки взаимной спектральной плотности исследована статистика вида

$$\tilde{f}_{ab}(\lambda) = \frac{2\pi}{T} \sum_{s=1}^T W_{ab}\left(\lambda - \frac{2\pi s}{T}\right) \hat{f}_{ab}\left(\frac{2\pi s}{T}\right), \quad (1)$$

где $W_{ab}(x)$, $x \in R$, $a, b = \overline{1, r}$ - спектральное окно, а $\hat{f}_{ab}^{(T)}(\lambda)$, $\lambda \in \Pi = [-\pi, \pi]$ - оценка взаимной спектральной плотности процесса $X(t)$, $t \in Z$, построенная по методу Уэлча [1]

$$\hat{f}_{ab}^{(T)}(\lambda) = \frac{1}{L} \sum_{l=1}^L I_{ab}(\lambda, l), \quad (2)$$

где

$$I_{ab}(\lambda, l) = \frac{1}{2\pi \sum_{p=0}^{N-1} h_a^N(p) h_b^N(p)} H_a(\lambda, l) \overline{H_b(\lambda, l)}, \quad (3)$$

$l = \overline{1, L}$, $a, b = \overline{1, r}$, $\lambda \in \Pi$, модифицированная периодограмма на l -ом интервале разбиения, а $H_a(\lambda, l)$ задано выражением

$$H_a(\lambda, l) = \sum_{t=0}^{N-1} h_a^N(t) X_a(t + (l-1)(N-K)) e^{-i\lambda(t+(l-1)(N-K))}, \quad (4)$$

$l = \overline{1, L}$, $a = \overline{1, r}$, $\lambda \in \Pi$, причем наблюдения сглаживаются одним и тем же окном просмотра данных $h_a^N(t)$, $t \in Z$.

В работе [1] исследована оценка (2) для гауссовских процессов. В данной работе оценки (1), (2) исследованы для произвольных случайных процессов.

Предположение 1. Пусть окна просмотра данных $h_a^N(t)$, $a = \overline{1, r}$, $t \in Z$, ограничены единицей и имеют ограниченную постоянную вариацию.

Предположение 2. Пусть $W_{ab}(x)$ непрерывная, периодическая функция с периодом 2π , имеет ограниченную вариацию и является ядром.

Теорема 1. Если взаимная спектральная плотность $f_{ab}(x)$ непрерывна в точке $x = \lambda$ и ограничена на Π , окна просмотра данных $h_a^N(t)$, $t \in R$, $a = \overline{1, r}$ удовлетворяют предположению 1, а спектральные окна предположению 2, то для оценки $\tilde{f}_{ab}(\lambda)$, $\lambda \in \Pi$, заданной выражением (1), справедливо соотношение

$$\lim_{T \rightarrow \infty} M\tilde{f}_{ab}(\lambda) = f_{ab}(\lambda), \quad (5)$$

$\lambda \in \Pi$, $a, b = \overline{1, r}$.

Доказательство. Используя свойства математического ожидания и функции $\Phi_{ab}(x)$, $x \in \Pi$, $a, b = \overline{1, r}$ вида

$$\Phi_{ab}(x) = [2\pi \sum_{t=0}^{N-1} h_a^N(t) h_b^N(t)]^{-1} \overline{\varphi_a(x) \varphi_b(x)},$$

где

$$\varphi_a(x) = \sum_{t=0}^{N-1} h_a^N(t) e^{itx},$$

запишем

$$\begin{aligned} I &= \left| M\tilde{f}_{ab}(\lambda) - f_{ab}(\lambda) \right| = \left| \frac{2\pi}{T} \sum_{s=1}^T W_{ab}\left(\lambda - \frac{2\pi s}{T}\right) M\tilde{f}_{ab}\left(\frac{2\pi s}{T}\right) - f_{ab}(\lambda) \right| = \\ &= \left| \int_{\Pi} f_{ab}(x) \frac{2\pi}{T} \sum_{s=1}^T W_{ab}\left(\lambda - \frac{2\pi s}{T}\right) \Phi_{ab}\left(x - \frac{2\pi s}{T}\right) dx - f_{ab}(\lambda) \right|. \end{aligned}$$

Откуда, учитывая лемму Д5.1 работы [2], получим

$$I = \left| \iint_{\Pi^2} W_{ab}(\lambda - \nu) \Phi_{ab}(x - \nu) f_{ab}(x) dx d\nu + O\left(\frac{1}{T}\right) - f_{ab}(\lambda) \right|.$$

Сделаем замену переменных $y = \lambda - \nu$, $x = x$, тогда,

$$I = \left| \iint_{\Pi^2} W_{ab}(y) \Phi_{ab}(x + y - \lambda) f_{ab}(x) dx dy - f_{ab}(\lambda) + O\left(\frac{1}{T}\right) \right|.$$

Сделаем замену переменных $z = x - \lambda$, $y = y$, получим

$$I = \left| \iint_{\Pi^2} W_{ab}(y) \Phi_{ab}(z + y) f_{ab}(z + \lambda) dz dy - f_{ab}(\lambda) + O\left(\frac{1}{T}\right) \right|.$$

Известно, что свертка двух ядер является ядром, следовательно

$$V_{ab}(z) = \int_{\Pi} W_{ab}(y) \Phi_{ab}(z + y) dy -$$

ядро. Тогда

$$\begin{aligned} I &= \left| \int_{\Pi} V_{ab}(z) f_{ab}(z + \lambda) dz - f_{ab}(\lambda) + O\left(\frac{1}{T}\right) \right| \leq \\ &\leq \int_{\Pi} |V_{ab}(z)| |f_{ab}(z + \lambda) - f_{ab}(\lambda)| dz + O\left(\frac{1}{T}\right). \end{aligned}$$

Так как взаимная спектральная плотность $f_{ab}(z)$ непрерывна в точке $z = \lambda$ и ограничена на Π , а $V_{ab}(z)$ является ядром, получим требуемый результат. Теорема доказана.

Таким образом, показано, что оценка взаимной спектральной плотности $\tilde{f}_{ab}(\lambda)$, $\lambda \in \Pi$, $a, b = \overline{1, r}$, заданная соотношением (1), является асимптотически несмещенной оценкой взаимной спектральной плотности процесса.

Построен график оценки спектральной плотности (2), полученной на основании данных температуры воздуха в г. Бресте за период с октября 2008 по февраль 2009 года для окон Дирихле, Фейера и Рисса.

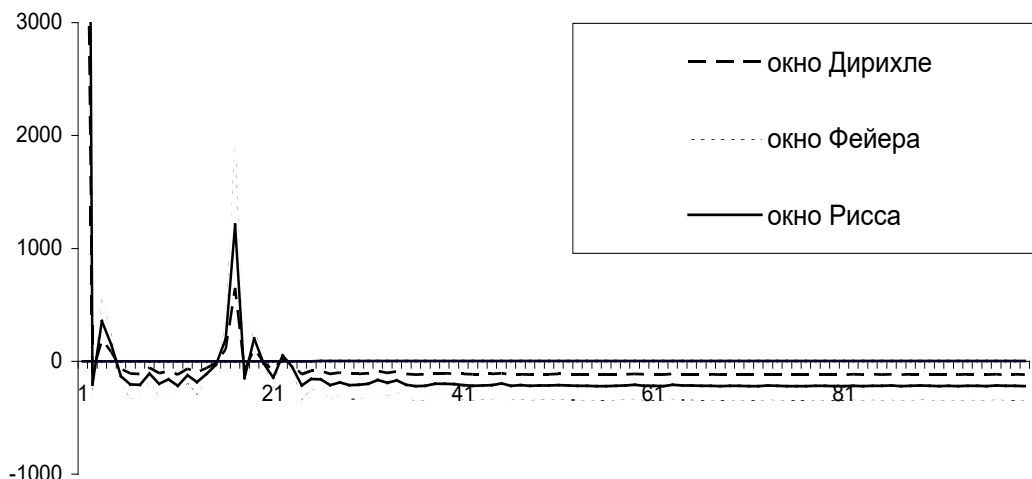


Рисунок 1 – Графики оценки спектральной плотности (1) для окон Дирихле, Фейера и Рисса

Литература

1. Welch, P.D. The use of FFT for the estimation of power spectra / P.D. Welch // IEEE Trans. Electroacoust. – 1967. – Vol. 15, №2. – P.70-73.
2. Бриллинджер, Д. Временные ряды. Обработка данных и теория / Д.Бриллинджер. – М.: Мир, 1980. – 536 с.

УДК 519.6 + 517.983.54

ОСТАНОВ ПО СОСЕДНИМ ПРИБЛИЖЕНИЯМ В МЕТОДЕ ИТЕРАЦИЙ ЯВНОГО ТИПА РЕШЕНИЯ НЕКОРРЕКТНЫХ ЗАДАЧ

Науомец С.Н.

УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Матысик О.В., канд. физ.-мат. наук, доцент

В гильбертовом пространстве H решается операторное уравнение $Ax = y_\delta$, где A – оператор положительный, ограниченный, несамосопряжённый и $\|y - y_\delta\| \leq \delta$. Предполагается, что $0 \in S_A$ (но не является собственным значением оператора A), поэтому рассматриваемая задача некорректна. Пусть $y \in R(A)$, т. е. при точной правой части y уравнение имеет единственное решение x . Будем искать его, используя явный итерационный метод

$$z_{n+1} = Cz_n + By_\delta + Cu_n, \quad z_0 \in H, \quad (1)$$

где
$$C = (E - \alpha A^* A)^2, \quad B = A^{-1} \left[E - (E - \alpha A^* A)^2 \right], \quad 0 < \alpha \leq \frac{5}{4 \|A^* A\|}, \quad \text{а}$$

u_n – ошибки вычисления итераций (причём $\|u_n\| \leq \beta$).

Предложенный метод можно сделать вполне эффективным, если воспользоваться следующим правилом останова *по соседним приближениям*: зададим уровень останова $\varepsilon > 0$ и момент останова m определим условиями

$$\|z_n - z_{n+1}\| > \varepsilon, \quad (n < m), \quad \|z_m - z_{m+1}\| \leq \varepsilon. \quad (2)$$

Для доказательства сходимости предложенного метода и получения оценки для момента останова необходимы следующие леммы.

Лемма 1. Пусть приближение w_n определяется равенством

$$w_0 = z_0, \quad w_{n+1} = Cw_n + By + Cu_n, \quad n \geq 0,$$

тогда справедливо неравенство

$$\sum_{k=0}^n \|w_k - w_{k+1} + Cu_k\|^2 \leq \|w_0 - x\|^2 + \sum_{k=0}^{n-1} \|Cu_k\|^2.$$

Лемма 2. При $\forall w_0 \in H$ и произвольной последовательности ошибок $\{u_n\}$, удовлетворяющих условию $\|u_n\| \leq \beta$, выполняется неравенство

$$\lim_{n \rightarrow \infty} \|w_n - w_{n+1}\| \leq 2\|C\|\beta.$$

Справедлива

Теорема. Пусть уровень останова $\varepsilon = \varepsilon(\delta, \beta)$ выбирается как функция от уровней δ и β норм погрешностей $y - y_\delta$ и u_n . Тогда справедливы следующие утверждения:

а) если $\varepsilon(\delta, \beta) > 2\|C\|\beta$, то момент останова m определён при любом начальном приближении $z_0 \in H$ и любых y_δ и u_n , удовлетворяющих условиям $\|y - y_\delta\| \leq \delta, \|u_n\| \leq \beta$;

б) если $\varepsilon(\delta, \beta) > \|B\|\delta + 2\|C\|\beta$, то тогда справедлива оценка

$$m \leq \frac{\|z_0 - x\|^2}{(\varepsilon - \|B\|\delta)(\varepsilon - \|B\|\delta - 2\|C\|\beta)};$$

в) если, кроме того, $\varepsilon(\delta, \beta) \rightarrow 0$, $\delta, \beta \rightarrow 0$ и $\varepsilon(\delta, \beta) \geq d(\|B\|\delta + \|C\|\beta^p)$, где $d > 1$, $p \in (0, 1)$, то $\lim_{\delta, \beta \rightarrow 0} \|z_m - x\| = 0$.

Для иллюстрации скоростных качеств метода (1) в гильбертовом пространстве $L_2(0, 1)$ решаем модельную задачу в виде уравнения:

$$\int_0^1 K(t, s)x(s)ds = y(t), \quad t \in [0, 1]$$

с симметричным положительным ядром $K(t, s) = \frac{1}{1 + 100(t - s)^2}$.

В качестве точного решения сформулированной задачи выберем функцию

$$x(s) = \begin{cases} s, & 0 \leq s < 0.5, \\ 1 - s, & 0.5 \leq s \leq 1. \end{cases}$$

Данная задача решалась методом (1) и хорошо известным в научной литературе методом простой итерации

$$x_{n+1, \delta} = x_{n, \delta} + \alpha(y_\delta - Ax_{n, \delta}), \quad x_{0, \delta} = 0. \quad (3)$$

Здесь воспользовались правилом останова (2), выбрав уровень останова $\varepsilon = 1,5\delta$. Итак, при $\delta = 10^{-3}$, $\varepsilon = 1,5 \cdot 10^{-3}$ для достижения оптимальной точности при счёте методом итераций (1) потребовалось 10 итераций, при счёте методом простой итерации (3) – 21 итерация. При $\delta = 10^{-4}$, $\varepsilon = 1,5 \cdot 10^{-4}$ соответственно потребовалось 17 и 48 итераций. Пример счёта показал, что для достижения оптимальной точности метод итераций (1) требует примерно в 2,5 раза меньше итераций, чем метод простой итерации (3).

УДК 519.6 + 517.983.54

К ВОПРОСУ О СХОДИМОСТИ ЯВНОГО ИТЕРАЦИОННОГО МЕТОДА РЕШЕНИЯ НЕКОРРЕКТНЫХ ЗАДАЧ

Наумовец С.Н.

*УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Матысик О. В., канд. физ.-мат. наук, доцент*

Пусть в гильбертовом пространстве H требуется решить уравнение

$$Ax = y \quad (1)$$

где A – ограниченный, самосопряженный, положительный оператор $A: H \rightarrow H$, для которого нуль не является собственным значением. Причем $0 \in S_A$, т. е. задача некорректна. Предполагается существование единственного решения x при точной правой части y . Для его отыскания предлагается итерационный метод

$$x_{n+1} = (E - \alpha A)^2 x_n + 2\alpha y - \alpha^2 Ay, \quad x_0 = 0. \quad (2)$$

Правую часть уравнения (1), как это обычно бывает на практике, считаем известной приближённо, т. е. вместо y известно δ -приближение y_δ , $\|y - y_\delta\| \leq \delta$. Тогда итерационный процесс (2) запишется в виде

$$x_{n+1, \delta} = (E - \alpha A)^2 x_{n, \delta} + 2\alpha y_\delta - \alpha^2 Ay_\delta, \quad x_{0, \delta} = 0. \quad (3)$$

Ниже, под сходимостью метода (3) понимается утверждение о том, что приближения (3) сколь угодно близко подходят к точному решению x уравнения (1) при подходящем выборе n и достаточно малых δ . Иными словами, метод итераций (3) является сходящимся, если $\lim_{\delta \rightarrow 0} \left(\inf_n \|x - x_{n,\delta}\| \right) = 0$.

В случае, когда нет сведений об истокообразной представимости точного решения ($x = A^s z, s > 0$), затруднительно получить априорные оценки погрешности и априорный момент останова. И, тем не менее, метод итераций (3) можно сделать вполне эффективным, если воспользоваться энергетической нормой гильбертова пространства $\|x\|_A = \sqrt{(Ax, x)}$, где $x \in H$. Покажем сходимость метода (3) в энергетической норме к решению (1) и получим для него априорные оценки погрешности в энергетической норме.

Рассмотрим разность $x - x_{n,\delta} = (x - x_n) + (x_n - x_{n,\delta})$. Запишем первое слагаемое в виде $x - x_n = A^{-1}(E - \alpha A)^{2n} y = (E - \alpha A)^{2n} x$.

Нетрудно показать, что $x - x_n \rightarrow 0$ в исходной норме гильбертова пространства H при $n \rightarrow \infty$, но скорость сходимости при этом может быть сколь угодно малой, и для её оценки необходимо предположить, что $x = A^s z, s > 0$. При использовании энергетической нормы нам это предположение не понадобится. Действительно, с помощью интегрального представления самосопряженного оператора $A = \int_0^M \lambda dE_\lambda$, где $M = \|A\|$ и E_λ – соответствующая спектральная функция, имеем

$$\|x - x_n\|_A^2 = \left(A(E - \alpha A)^{2n} x, (E - \alpha A)^{2n} x \right) = \int_0^M \lambda (1 - \alpha\lambda)^{4n} d(E_\lambda x, x),$$

$$\|x_n - x_{n,\delta}\|_A^2 = \int_0^M \lambda^{-1} \left[1 - (1 - \alpha\lambda)^{2n} \right]^2 d(E_\lambda (y - y_\delta), y - y_\delta).$$

Оценив подынтегральные функции, получим при условии $0 < \alpha \leq \frac{5}{4M}$ оценку погрешности для итерационного метода (3) в энергетической норме

$$\|x - x_{n,\delta}\|_A \leq (4n\alpha e)^{-1/2} \|x\| + \left(\frac{35}{27} n\alpha \right)^{1/2} \delta, \quad n \geq 1. \quad (4)$$

Таким образом, переход к энергетической норме как бы заменяет предположение об истокопредставимости порядка $s = \frac{1}{2}$ для точного решения. Более того, для сходимости

$\|x - x_{n,\delta}\|_A \rightarrow 0, n \rightarrow \infty$, достаточно, чтобы $\sqrt{n}\delta \rightarrow 0, n \rightarrow \infty, \delta \rightarrow 0$.

Оптимизируем оценку (4) по n . Для этого при заданном δ найдём такое значение числа итераций n , при котором оценка погрешности становится минимальной. Приравняв нулю производную по n от правой части неравенства (4), получим $n_{\text{opt}} = (35/27)^{-1/2} (2\alpha)^{-1} e^{-1/2} \delta^{-1} \|x\|$. Подставив n_{opt} в оценку (4), найдём её оптимальное значение

$$\|x - x_{n,\delta}\|_A^{\text{opt}} \leq (35/27)^{1/4} (2\delta\|x\|)^{1/2} e^{-1/4}. \quad (5)$$

Замечание. Из неравенства (5) вытекает, что оптимальная оценка погрешности не зависит от параметра α . Но n_{opt} зависит от α и, поэтому для уменьшения n и, значит, объема вычислительной работы следует брать α возможно большим, удовлетворяющим условию $0 < \alpha \leq \frac{5}{4M}$, и так, чтобы n_{opt} было целым.

УДК 517.512.2

АНАЛОГ ЗАДАЧИ КОШИ ДЛЯ НЕЛИНЕЙНОГО ДИФФЕРЕНЦИАЛЬНОГО УРАВНЕНИЯ СО СТАРШЕЙ ПРОИЗВОДНОЙ ДРОБНОГО ПОРЯДКА

Онищук А.И.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

В работе [1] указаны условия существования и единственности решения аналога задачи Коши для дифференциального уравнения

$$y^{(\alpha)} = f(x, y), \quad (1)$$

где $0 < \alpha < 1$, найдена оценка приближения решения уравнения (1) решением специально построенного, с помощью линейных методов суммирования интегралов Фурье, операторного уравнения типа Абеля-Гаммерштейна.

Пусть Π параллелепипед в \mathbf{R}^4 вида:

$$\Pi := \{(x, y^{(0)}, y^{(1)}, y^{(2)}) \in \mathbf{R}^4 \mid 0 \leq x \leq l, y_0^{(j)} - h_j \leq y^{(j)} \leq y_0^{(j)} + h_j, j = \overline{0, 2}\},$$

где $l, h_j \in \mathbf{R}_+$ ($j = \overline{0, 2}$), $f : \Pi \mapsto \mathbf{R}$ заданная, абсолютно интегрируемая по параллелепипеду Π функция.

Через $L_f^{(2)}(0; l)$ обозначим класс дважды дифференцируемых на отрезке $[0; l]$ функций $y = y(x)$ с абсолютно непрерывной производной y'' , и таких, что функция $f(x, y(x), y'(x), y''(x), y'''(x))$ абсолютно непрерывна на отрезке $[0, l]$. Нетрудно видеть, что при задании нормы формулой

$$\|y\| := \sum_{j=0}^3 \int_0^l |y^{(j)}(x)| dx,$$

$L_f^{(2)}(0; l)$ является банаховым пространством.

Рассмотрим задачу нахождения функции $y = y(x)$ класса $L_f^{(2)}(0; l)$, удовлетворяющей нелинейному дифференциальному уравнению, с дробной старшей производной

$$y^{(\alpha)} = f(x, y, y', y'', y'''), \quad 2 < \alpha < 3, \quad (2)$$

и начальным условиям

$$y^{(\alpha-1)}(0) = 0, \quad y^{(\alpha-2)}(0) = 0, \quad y^{(\alpha-2)}(0) = 0. \quad (3)$$

Теорема. Пусть функция $f : \Pi \mapsto \mathbf{R}$ абсолютно интегрируема на параллелепипеде Π и существует $A > 0$, что для любых двух точек $M_1(x, y_1^{(0)}, y_1^{(1)}, y_1^{(2)}, y_1^{(3)})$ и $M_2(x, y_2^{(0)}, y_2^{(1)}, y_2^{(2)}, y_2^{(3)})$ из Π выполняется неравенство

$$|f(M_1) - f(M_2)| \leq A \sum_{j=0}^3 |y_1^{(j)} - y_2^{(j)}|.$$

Если

$$A(l^\alpha / \alpha + \alpha(\alpha - 1)(\alpha - 2)l^{\alpha-2}(1 + (\alpha - 3)l^{\alpha-3}) + \alpha(\alpha - 1)l^{\alpha-1}) / \Gamma(\alpha) < 1,$$

то задача (2), (3) имеет в классе $L_f^{(2)}(0;l)$ единственное решение.

Литература

1. Семенчук, Н.П. Дифференциальные уравнения / Н.П. Семенчук 1982. – Т. 18. – № 10. – С. 1831–1833.

УДК 517

ОБ ОДНОМ КЛАССЕ НЕЛИНЕЙНЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ СО СТАРШЕЙ ПРОИЗВОДНОЙ НЕЦЕЛОГО ПОРЯДКА

Охримчук А.В., Дацьк В.Т.

УО «Брестский государственный технический университет», г. Брест

Найдены условия существования и единственности задачи Коши и получена оценка приближения решения для дифференциального уравнения нецелого порядка.

$$y^{(\alpha)} = f(x, y, y', \dots, y^{(n-1)}), \quad (1)$$

где $n-1 < \alpha < n, n \in \mathbb{N}$.

Пусть функция f - абсолютно интегрируема на параллелепипеде $\Pi_{l_{nj}}$.

Решение уравнения (1) будем искать в классе дифференцируемых до порядка $n-1, n \in \mathbb{N}$, функций $y = y(x)$ на отрезке $[0, l]$ с абсолютно непрерывной на этом отрезке производной $y^{(n-1)}(x)$. Причем, для любых указанных функций $y = y(x)$ функция $\mu(x) = f(x, y(x), y'(x), \dots, y^{(n-1)}(x))$ должна быть также абсолютно непрерывной на отрезке $[0, l]$ (условие (*)). Норма для функций $y = y(x)$ вводится по формуле:

$$\|y(x)\| = \sum_{j=0}^{n-1} \int_0^l |y^{(j)}(x)| dx \quad (2)$$

Указанный класс функций $y = y(x)$ обозначим через $L^{(j)}(0, l)$.

Теорема 1. Если для уравнения (1) функция f — абсолютно интегрируема на параллелепипеде $\Pi_{l_{nj}}$ и для любых точек $M_1(x, y_1, y_1', \dots, y_1^{(n-1)})$ и $M_2(x, y_2, y_2', \dots, y_2^{(n-1)})$ из $\Pi_{l_{nj}}$ будет

$$|f(M_1) - f(M_2)| \leq A \sum_{j=0}^{n-1} |y_1^{(j)} - y_2^{(j)}|, \quad (3)$$

где A — некоторая положительная константа, а также выполняется условие (*), то уравнение (1) при выполнении неравенства

$$\frac{A}{\bar{A}(\alpha)} \left(\frac{l^\alpha}{\alpha} + \sum_{j=1}^{n-1} (\alpha - 1)(\alpha - 2) \dots (\alpha - (j - 1)) l^{\alpha-j} \right) < 1$$

имеет в классе $L^{(j)}(0, l)$ единственное решение $y = y(x)$, удовлетворяющее начальным условиям

$$\bar{A}^{-(1-\alpha)} y(0) = y_0^{(n-1)} = \bar{A}^{-(2-\alpha)} y(0) = y_0^{(n-2)} = \dots = \bar{A}^{-(n-\alpha)} y(0) = y_0 = 0, \quad (4)$$

где Γ - гамма-функция, $D^{-\beta}$ - дробный интеграл порядка $\beta > 0$, D^β - дробная производная порядка $\beta > 0$.

О ФУНДАМЕНТАЛЬНОМ РЕШЕНИИ ОПЕРАТОРА Δ^2 В \mathbb{R}^3

Савчук Л.А.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
 Научный руководитель – Басик А.И., преподаватель

Понятие фундаментального решения является одним из основных понятий теории уравнений с частными производными. В настоящей работе указывается фундаментальное решение оператора Δ^2 в \mathbb{R}^3 , где $\Delta = \partial^2/\partial x_1^2 + \partial^2/\partial x_2^2 + \partial^2/\partial x_3^2$ – оператор Лапласа, $\Delta^2 u = \Delta(\Delta u)$. Далее, с его помощью, выводится интегральное представление функций класса $C^4(\overline{\Omega})$ и доказываются некоторые свойства бигармонических функций – решений уравнения $\Delta^2 u = 0$.

Пусть $\Omega \subset \mathbb{R}^3$ ограниченная область с кусочно-гладкой границей $\partial\Omega$. Для функций $u = u(x)$ и $v = v(x)$ класса $C^2(\overline{\Omega})$ известна вторая формула Грина (см. [1], стр. 69)

$$\int_{\Omega} u \Delta v dx = \int_{\Omega} v \Delta u dx + \int_{\partial\Omega} \left(u \frac{\partial v}{\partial \nu} - v \frac{\partial u}{\partial \nu} \right) dS, \quad (1)$$

где $\nu = \nu(x) = (\nu_1(x), \nu_2(x), \nu_3(x))$ – единичное поле внешних нормалей на поверхности $\partial\Omega$.

Если $u, v \in C^4(\overline{\Omega})$, то заменяя в формуле (1) v на Δv , получим

$$\int_{\Omega} u \Delta^2 v dx = \int_{\Omega} \Delta v \Delta u dx + \int_{\partial\Omega} \left(u \frac{\partial \Delta v}{\partial \nu} - \Delta v \frac{\partial u}{\partial \nu} \right) dS, \quad (2)$$

затем, заменяя в формуле (2) u на v и v на u , получим

$$\int_{\Omega} v \Delta^2 u dx = \int_{\Omega} \Delta v \Delta u dx + \int_{\partial\Omega} \left(v \frac{\partial \Delta u}{\partial \nu} - \Delta u \frac{\partial v}{\partial \nu} \right) dS. \quad (3)$$

Вычитая из (2) формулу (3), получим аналог второй формулы Грина для оператора Δ^2 :

$$\int_{\Omega} u \Delta^2 v dx = \int_{\Omega} v \Delta^2 u dx + \int_{\partial\Omega} \left(\left(u \frac{\partial \Delta v}{\partial \nu} + \Delta u \frac{\partial v}{\partial \nu} \right) - \left(v \frac{\partial \Delta u}{\partial \nu} + \Delta v \frac{\partial u}{\partial \nu} \right) \right) dS. \quad (4)$$

Определение 1. Функция $E(x, y)$ называется фундаментальным решением оператора Δ^2 в \mathbb{R}^3 , если при каждом $y \in \mathbb{R}^3$ $E(x, y)$ является обобщенной функцией из пространства $D'(\mathbb{R}^3)$ и удовлетворяет уравнению $\Delta^2 E = \delta(x - y)$, т.е. для каждой функции $\varphi \in C_0^\infty(\mathbb{R}^3)$ выполняется равенство:

$$\langle \Delta^2 E(x, y), \varphi(x) \rangle = \varphi(y). \quad (5)$$

Теорема 2. Функция $E(x, y) = -\frac{1}{8\pi} r$ является фундаментальным решением оператора Δ^2 , где $r = |x - y| = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2}$.

Доказательство. Зафиксируем произвольную точку $y \in \mathbb{R}^3$. Для каждой финитной бесконечно дифференцируемой функции $\varphi(x)$ в \mathbb{R}^3 , выберем число $R > 0$ так, что носитель $\text{supp } \varphi(x)$ функции $\varphi(x)$ лежит в шаре $B(y, R)$ с центром в точке y радиуса R .

Поскольку функция $E(x, y)$ непрерывна по x , то она локально интегрируема по x и, следовательно,

$$\begin{aligned} \langle \Delta^2 E, \varphi \rangle &:= \langle E, \Delta^2 \varphi \rangle = \int_{\mathbb{R}^3} E(x, y) \Delta^2 \varphi(x) dx = \\ &= \int_{|x-y| < R} E(x, y) \Delta^2 \varphi(x) dx = \lim_{\varepsilon \rightarrow +0} \int_{\varepsilon < |x-y| < R} E(x, y) \Delta^2 \varphi(x) dx. \end{aligned}$$

Преобразуем последний интеграл с помощью аналога второй формулы Грина (4). Выбрав $u = E(x, y)$ и $v = \varphi(x)$, получим

$$\begin{aligned} \int_{\varepsilon < |x-y| < R} E(x, y) \Delta^2 \varphi(x) dx &= \int_{|x-y|=\varepsilon} E(x, y) \frac{\partial \Delta \varphi(x)}{\partial v_x} dS(x) + \int_{|x-y|=\varepsilon} \Delta E(x, y) \frac{\partial \varphi(x)}{\partial v_x} dS(x) - \\ &- \int_{|x-y|=\varepsilon} \varphi(x) \frac{\partial \Delta E(x, y)}{\partial v_x} dS(x) - \int_{|x-y|=\varepsilon} \Delta \varphi(x) \frac{\partial E(x, y)}{\partial v_x} dS(x). \end{aligned} \quad (6)$$

Поскольку $v = \frac{y-x}{\varepsilon}$ – вектор единичной нормали к поверхности $|x-y| = \varepsilon$, внешней для области $\varepsilon < |x-y| < r$, то при $|x-y| = \varepsilon$ справедливы формулы

$$\frac{\partial E(x, y)}{\partial v_x} = \frac{1}{8\pi}, \quad \Delta E(x, y) = -\frac{1}{4\pi\varepsilon}, \quad \frac{\partial \Delta E(x, y)}{\partial v_x} = -\frac{1}{4\pi\varepsilon^2}.$$

Отсюда следует, что при $\varepsilon \rightarrow +0$ первый, второй и четвёртый интегралы в формуле (6) стремятся к нулю. Для третьего интеграла в (6) имеем:

$$\begin{aligned} - \int_{|x-y|=\varepsilon} \varphi(x) \frac{\partial \Delta E(x, y)}{\partial v_x} dS(x) &= \frac{1}{4\pi\varepsilon^2} \int_{|x-y|=\varepsilon} \varphi(x) dS(x) = \\ &= \frac{1}{4\pi\varepsilon^2} \cdot \varphi(x_\varepsilon) \cdot 4\pi\varepsilon^2 = \varphi(x_\varepsilon) \xrightarrow{\varepsilon \rightarrow +0} \varphi(y). \end{aligned}$$

Здесь мы применили известную теорему о среднем для интеграла от непрерывной функции

$$\int_{|x-y|=\varepsilon} \varphi(x) dS(x) = 4\pi\varepsilon^2 \varphi(x_\varepsilon),$$

где x_ε – некоторая точка на сфере с центром в точке y и радиуса ε .

Таким образом, соотношение (5) выполнено. Что и требовалось доказать. Аналогично доказывается следующее утверждение.

Теорема 3. (Интегральное представление функций класса $C^4(\overline{\Omega})$) Пусть $\Omega \subset \mathbb{R}^3$ – ограниченная область с кусочно-гладкой границей $\partial\Omega$, функция $u \in C^4(\overline{\Omega})$, тогда в каждой точке $y \in \Omega$ справедливо равенство

$$u(y) = \int_{\Omega} E(x, y) \Delta^2 u dx + \int_{\partial\Omega} \left(\left(u \frac{\partial \Delta E}{\partial v_x} + \Delta u \frac{\partial E}{\partial v_x} \right) - \left(E \frac{\partial \Delta u}{\partial v_x} + \Delta E \frac{\partial u}{\partial v_x} \right) \right) dS(x). \quad (7)$$

Следствие. (Интегральное представление бигармонических функций) Пусть $\Omega \subset \mathbb{R}^3$ – ограниченная область с кусочно-гладкой границей, функция $u \in C^4(\overline{\Omega})$ и $\Delta^2 u = 0$ в Ω . Тогда

$$u(y) = \int_{\partial\Omega} \left(\left(u \frac{\partial \Delta E}{\partial \nu} + \Delta u \frac{\partial E}{\partial \nu} \right) - \left(E \frac{\partial \Delta u}{\partial \nu} + \Delta E \frac{\partial u}{\partial \nu} \right) \right) dS \quad (8)$$

в каждой точке $y \in \Omega$; если же $\Delta^2 u = f(x)$ в Ω , то

$$u(y) = \int_{\Omega} E(x, y) f(x) dx + \int_{\partial\Omega} \left(\left(u \frac{\partial \Delta E}{\partial \nu} + \Delta u \frac{\partial E}{\partial \nu} \right) - \left(E \frac{\partial \Delta u}{\partial \nu} + \Delta E \frac{\partial u}{\partial \nu} \right) \right) dS.$$

Укажем теперь некоторые свойства бигармонических функций. Положив в формуле Грина (4) $\nu \equiv 1$, получим следующее утверждение.

Теорема 4. Пусть функция $u \in C^4(\overline{\Omega})$ и $\Delta^2 u = 0$ в Ω , тогда имеет место равенство:

$$\int_{\partial\Omega} \frac{\partial \Delta u}{\partial \nu} dS(x) = 0. \quad (9)$$

Теорема 5. (О среднем арифметическом по сфере) Пусть функция $u \in C^4(B(y, R)) \cap C^3(B[y, R])$ и $\Delta^2 u = 0$ в $B(y, R)$, тогда имеет место равенство:

$$u(y) = \frac{1}{4\pi \cdot R^2} \int_{|x-y|=R} u(x) dS + \frac{1}{4\pi \cdot R} \int_{|x-y|=R} \frac{\partial u}{\partial \nu} dS - \frac{R^2}{2} \Delta u(y). \quad (10)$$

Доказательство. Выберем произвольно число $r \in (0, R)$, тогда функция $u \in C^4(B[y, r])$. По следствию из теоремы 3 имеет место равенство

$$\begin{aligned} u(y) &= \int_{|x-y|=r} \left(\left(u \frac{\partial \Delta E}{\partial \nu} + \Delta u \frac{\partial E}{\partial \nu} \right) - \left(E \frac{\partial \Delta u}{\partial \nu} + \Delta E \frac{\partial u}{\partial \nu} \right) \right) dS = \\ &= \frac{1}{4\pi \cdot r^2} \int_{|x-y|=r} u(x) dS(x) - \frac{1}{8\pi} \int_{|x-y|=r} \Delta u dS(x) + \\ &+ \frac{1}{4\pi \cdot r} \int_{|x-y|=r} \frac{\partial u(x)}{\partial \nu} dS(x) + \frac{1}{8\pi} r \int_{|x-y|=r} \frac{\partial \Delta u}{\partial \nu} dS(x). \end{aligned}$$

Учитывая (9) и перейдя в последнем равенстве к пределу при $r \rightarrow R$, получим:

$$u(y) = \frac{1}{4\pi R^2} \int_{|x-y|=R} u(x) dS + \frac{1}{4\pi R} \int_{|x-y|=R} \frac{\partial u}{\partial \nu} dS - \frac{1}{8\pi} \int_{|x-y|=R} \Delta u dS. \quad (11)$$

Так как $\Delta(\Delta u) = 0$, то по теореме «О среднем значении по сфере» (см. [1], стр. 77) для оператора Δ имеет место равенство:

$$\Delta u(y) = \frac{1}{4\pi R^2} \int_{|x-y|=R} \Delta u(x) dS(x).$$

Подставляя последний интеграл в (11), получим формулу (10).

Литература

1. Олейник, О.А. Лекции об уравнениях с частными производными / О.А. Олейник. – М.: БИНОМ, 2005.

УДК 517.925

К ВОПРОСУ ИНТЕГРИРУЕМОСТИ В КВАДРАТУРАХ ЛИНЕЙНОГО ДИФФЕРЕНЦИАЛЬНОГО УРАВНЕНИЯ ВТОРОГО ПОРЯДКА С ШЕСТЬЮ ОСОБЫМИ ТОЧКАМИ

Степанюк Г.П.

Волынский национальный университет имени Леси Украинки, г. Луцк, Украина

Чичурин А.В.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

Известно, что линейное дифференциальное уравнение второго порядка

$$y'' + p(z)y' + q(z)y = 0, \tag{1}$$

с произвольными аналитическими коэффициентами $p(z)$ и $q(z)$ не интегрируется в квадратурах в общем виде (например, [1]). Имеется, однако, много конкретных примеров функций $p(z)$ и $q(z)$, когда уравнение (1) интегрируется в квадратурах, либо определяет специальные функции.

Наша задача – указать способ интегрирования уравнения (1) в квадратурах, когда коэффициенты $p(z)$ и $q(z)$ являются дробно-рациональными функциями и имеют шесть полюсов по z . Коэффициенты $p(z)$ и $q(z)$ в явной форме запишем в виде

$$p(z) = \frac{\sum_{i=0}^4 \beta_i z^{4-i}}{z \prod_{k=1}^4 (z - a_k)}, \quad q(z) = \frac{\sum_{i=0}^4 \gamma_i z^{4-i}}{z^2 \prod_{k=1}^4 (z - a_k)^2}, \tag{2}$$

где a_k ($k = \overline{1,4}$), β_i ($i = \overline{0,4}$), γ_j ($j = \overline{0,8}$) – постоянные. Особыми точками уравнения (1), (2) являются точки $0, a_1, a_2, a_3, a_4, \infty$. Важность исследования уравнений (1) с числом особых точек больших трех, отмечена в работах [2, 3].

Для решения сформулированной задачи воспользуемся методом, приведенным в работе [4]. Согласно этому методу, требуется найти частное решение уравнения Риккати

$$2b' = b^2 - 2p(z)b + 4q(z), \tag{3}$$

где коэффициенты $p(z)$, $q(z)$ определяются соотношениями (2).

Частное решение уравнения (3) будем искать в виде

$$b(z) = \frac{\sum_{i=0}^4 \delta_i z^{4-i}}{z \prod_{k=1}^4 (z - a_k)} \tag{4}$$

где δ_i ($i = \overline{0,4}$) – постоянные, подлежащие определению.

Подставляя (4) в уравнение (3), (2), получим систему из девяти уравнений вида

$$\begin{aligned}
 4\gamma_0 + \delta_0(2 - 2\beta_0 + \delta_0) &= 0, \\
 4\gamma_1 - 2\beta_1\delta_0 + 2(2 - \beta_0 + \delta_0)\delta_1 &= 0, \\
 4\gamma_2 - 2a_2a_3\delta_0 - 2a_2a_4\delta_0 - 2a_3a_4\delta_0 - 2\beta_2\delta_0 - 2a_2\delta_1 - 2a_3\delta_1 - 2a_4\delta_1 - \\
 - 2\beta_1\delta_1 + \delta_1^2 - 2a_1(a_2\delta_0 + a_3\delta_0 + a_4\delta_0 + \delta_1) + 6\delta_2 - 2\beta_0\delta_2 + 2\delta_0\delta_2 &= 0, \\
 2(2\gamma_3 + 2a_2a_3a_4\delta_0 - \beta_3\delta_0 - \beta_2\delta_1 + 2a_1(a_3a_4\delta_0 + a_2(a_3 + a_4)\delta_0 - \delta_2) - \\
 - 2a_2\delta_2 - 2a_3\delta_2 - 2a_4\delta_2 - \beta_1\delta_2 + \delta_1\delta_2 + 4\delta_3 - \beta_0\delta_3 + \delta_0\delta_3) &= 0, \\
 4\gamma_4 - 2\beta_4\delta_0 + 2a_2a_3a_4\delta_1 - 2\beta_3\delta_1 + 2a_2a_3\delta_2 + 2a_2a_4\delta_2 + 2a_3a_4\delta_2 + \\
 + 2\beta_2\delta_2 + \delta_2^2 + 2a_1(a_4\delta_2 + a_3(a_4\delta_1 + \delta_2)) + a_2(a_4\delta_1 + a_3(-3a_4\delta_0 + \\
 + \delta_1) + \delta_2) - 3\delta_3) - 6a_2\delta_3 - 6a_3\delta_3 - 6a_4\delta_3 - 2\beta_1\delta_3 + 2\delta_1\delta_3 + 10\delta_4 - \\
 - 2\beta_0\delta_4 + 2\delta_0\delta_4 &= 0, \\
 - 2(-2\gamma_5 + \beta_4\delta_1 + \beta_3\delta_2 - 2a_2a_3\delta_3 - 2a_2a_4\delta_3 - 2a_3a_4\delta_3 + \beta_2\delta_3 - \delta_2\delta_3 + \\
 + 4a_2\delta_4 + 4a_3\delta_4 + 4a_4\delta_4 + \beta_1\delta_4 - \delta_1\delta_4 + 2a_1(a_2(a_3a_4\delta_1 - \delta_3) - a_3\delta_3 - \\
 - a_4\delta_3 + 2\delta_4)) &= 0, \\
 4\gamma_6 - 2\beta_4\delta_2 - 2a_2a_3a_4\delta_3 - 2\beta_3\delta_3 + \delta_3^2 + 6a_2a_3\delta_4 + 6a_2a_4\delta_4 + \\
 + 6a_3a_4\delta_4 - 2\beta_2\delta_4 + 2\delta_2\delta_4 - 2a_1(a_3(a_4\delta_3 - 3\delta_4) + a_2(a_4\delta_3 + a_3(a_4\delta_2 + \\
 + \delta_3) - 3\delta_4) - 3a_4\delta_4) &= 0, \\
 - 2(-2\gamma_7 + \beta_4\delta_3 + (2a_2a_3a_4 + 2a_1(a_3a_4 + a_2(a_3 + a_4))) + \beta_3 - \delta_3)\delta_4) &= 0, \\
 4\gamma_8 + \delta_4(2a_1a_2a_3a_4 - 2\beta_4\delta_4) &= 0.
 \end{aligned} \tag{5}$$

Система (5) является алгебраической системой относительно неизвестных, где $\delta_i (i = \overline{0,4})$. Уравнения (5₁) и (5₉) являются квадратными относительно δ_0 и δ_4 соответственно; уравнения (5₂) – (5₄) – линейные относительно $\delta_1 - \delta_3$ соответственно.

Таким образом, найдя корни уравнений (5₁), (5₉), и (5₂) – (5₄), подставим найденные значения $\delta_i (i = \overline{0,4})$ в четыре оставшихся уравнения, которые определяют соотношения между коэффициентами $\beta_i (i = \overline{0,4})$, $\gamma_j (j = \overline{0,8})$ и полюсами $a_k (k = \overline{1,4})$.

В качестве примера приведем решение системы (5) для следующих значений полюсов $a_1 = 1, a_2 = 2, a_3 = 3, a_4 = 4$ и коэффициентов

$$\begin{aligned}
 \beta_0 = 1, \beta_1 = 1, \beta_2 = 105, \beta_3 = -100, \beta_4 = 24, \gamma_0 = -1, \gamma_1 = 1, \gamma_2 = \beta_2 + 35, \\
 \gamma_3 = \beta_3 - 100, \gamma_4 = (576 - 48\beta_4 + \beta_4^2)/4: \gamma_4 = 96, \gamma_5 = 0, \gamma_6 = 0, \gamma_7 = 0, \\
 \delta_0 = 2, \delta_1 = 0, \delta_2 = 0, \delta_3 = 0, \delta_4 = 4.
 \end{aligned}$$

Таким образом, частное решение (4) примет вид

$$b(z) = \frac{2z^3}{(z-1)(z-2)(z-3)(z-4)} \tag{6}$$

Коэффициенты $p(z)$ и $q(z)$ вида (2) уравнения (3) примут вид

$$p(z) = \frac{z^4 + z^3 + 105z^2 - 100z + 24}{z(z-1)(z-2)(z-3)(z-4)},$$

$$q(z) = \frac{-z^8 + z^7 + 145z^6 - 200z^5 + 96z^4}{z^2(z-1)^2(z-2)^2(z-3)^2(z-4)^2}.$$
(7)

Зная коэффициенты (7) уравнения Риккати (3) и его частное решение (6) легко найти его общее решение. Тогда, согласно работе [4], найдем общее решение уравнения (1), (2) в виде

$$y = C_1 \frac{e^{\frac{1}{2} \int b(z) dz}}{e^{\int b(z) dz}} (C_1 - \int e^{\int b(z) dz} C_2 e^{-\int b(z) dz} dz)$$
(8)

где C_1, C_2 – произвольные постоянные.

Для приведенного примера величина $\int b(z) dz$ в соотношении (8) равна

$$\int \frac{2z^3}{z(z-1)(z-2)(z-3)(z-4)} dz = \frac{64}{3} \ln(z-4) - 27 \ln(z-3) + 8 \ln(z-2) - \frac{1}{3} \ln(z-1).$$

Литература

1. Матвеев, Н.М. Методы интегрирования обыкновенных дифференциальных уравнений / Н.М. Матвеев – СПб: Изд-во «Лань», 2003.
2. Славянов, С.Ю. Специальные функции: Единая теория, основанная на анализе особенностей / С.Ю. Славянов, В. Лай – СПб.: Невский Диалект, 2002. – 312 с.
3. Уиттекер, Э.Т. Курс современного анализа / Э.Т. Уиттекер, Дж.Н. Ватсон – М.: Физматгиз, 1963. – Ч.1. – 344 с.
4. Лукашевич, Н.А. Об интегрируемости в квадратурах линейных дифференциальных уравнений второго порядка / Н.А. Лукашевич, А.В. Чичурин // Веснік Брэсцкага ун-та. – 2006. – № 2 (26). – С. 11 - 15.

УДК 517.988.6

О ДВУХПАРАМЕТРИЧЕСКИХ КВАЗИНЬЮТОНОВСКИХ ИТЕРАЦИОННЫХ МЕТОДАХ ДЛЯ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

Стрилец Н.Н.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

В работе [1] для решения уравнения

$$f(x) = 0, \quad f : D \subset X \rightarrow X, \quad X - B\text{-пространство}$$

предложен двухпараметрический квазиньютоновский итерационный метод, сходящийся со сверхлинейной (локально квадратичной) скоростью.

Приведем алгоритм реализации метода, который отличается от метода в [1] лишь некоторыми деталями при вычислении шаговой длины.

На нулевой итерации задан начальный набор параметров (x_0, β_0, γ_0) , где $\beta_0 \in [10^{-4}, 1]$, $\gamma_0 = \beta_0^2$.

На n -ой итерации ($n=1,2,\dots$) уже имеется набор параметров $(x_{n-1}, \beta_{n-1}, \gamma_{n-1})$, а очередной набор (x_n, β_n, γ_n) вычисляется в результате поэтапного выполнения следующей последовательности действий.

Этап 1. Находится поправка из линейного уравнения

$$f'(x_{n-1})\Delta x_{n-1} = -f(x_{n-1}), \quad (1)$$

где $f'(x_n)$ — производная Фреше оператора f .

Этап 2. Следующее приближение вычисляется по формуле:

$$x_n := x_{n-1} + \beta_{n-1}\Delta x_{n-1}. \quad (2)$$

Этап 3. Если $\|f(x_n)\| < \varepsilon$, где ε — точность, то выход из итерационного процесса, иначе переход к следующему шагу.

Этап 4. Определяется новая шаговая длина: если $\beta_{n-1} = 1$, то $\beta_n := 1$, иначе

$$\beta_n := \min\left(1, \frac{\gamma_{n-1}\|f(x_{n-1})\|}{\beta_{n-1}\|f(x_n)\|}\right), \quad \gamma_n := \frac{\gamma_{n-1}\|f(x_{n-1})\|}{\|f(x_n)\|} \quad (3)$$

и осуществляется переход к следующей итерации.

Квазиньютоновский итерационный метод, алгоритм реализации которого описывается этапами 1–4, обозначим как процесс (1)–(3).

Сходимость итерационного процесса (1)–(3) при различных ограничениях на гладкость оператора f доказывается практически аналогично доказательствам соответствующих результатов из работы [2], в которой рассматривается так называемый полностью регуляризованный вариант данного процесса.

Процесс (1)–(3) допускает альтернативное представление. Для этого введем следующую формулу пересчета шаговой длины:

$$\beta_n = \begin{cases} \min\left(1, \frac{\|f(x_0)\|}{\|f(x_n)\|}\beta_0\right), & n=1, \\ \min\left(1, \frac{\|f(x_{n-1})\|}{\|f(x_n)\|}\beta_{n-2}\right), & n=2,3,\dots \end{cases} \quad (4)$$

Итерационный процесс, который получается из процесса (1)–(3) заменой формулы (3) на формулу (4), обозначим как процесс (1),(2),(4).

Теорема 1. Итерационные процессы (1)–(3) и (1),(2),(4) при одинаковых начальных параметрах (x_0, β_0) эквивалентны (т.е. генерируют одну и ту же последовательность приближений $\{x_n\}$).

Доказательство. Рассмотрим итерационный процесс (1)–(3). Очевидно, что найдутся n такие, что $\beta_n < 1$. Для них из (3) последовательно получим соотношения

$$\begin{aligned} \beta_0, & \quad \gamma_0 = \beta_0^2, \\ \beta_1 = \frac{\beta_0\|f(x_0)\|}{\|f(x_1)\|}, & \quad \gamma_1 = \frac{\gamma_0\|f(x_0)\|}{\|f(x_1)\|} = \beta_0 \frac{\beta_0\|f(x_0)\|}{\|f(x_1)\|} = \beta_1\beta_0, \\ \beta_2 = \frac{\gamma_1\|f(x_1)\|}{\beta_1\|f(x_2)\|} = \frac{\beta_0\|f(x_1)\|}{\|f(x_2)\|}, & \quad \gamma_2 = \frac{\gamma_1\|f(x_1)\|}{\|f(x_2)\|} = \beta_1\beta_0 \frac{\beta_2}{\beta_0} = \beta_2\beta_1. \end{aligned}$$

.....

Учитывая полученные соотношения для β_n , индуктивным путем придем (4). В результате, мы получили итерационный процесс (1),(2),(4).

Докажем теперь обратный переход. Рассмотрим процесс (1),(2),(4). Очевидно, что найдутся n такие, что $\beta_n < 1$. Для них из (4) будем иметь

$$\beta_n = \frac{\|f(x_{n-1})\|}{\|f(x_n)\|} \beta_{n-2} = \frac{\beta_{n-1} \beta_{n-2} \|f(x_{n-1})\|}{\beta_{n-1} \|f(x_n)\|} = \frac{\gamma_{n-1} \|f(x_{n-1})\|}{\beta_{n-1} \|f(x_n)\|}, \quad (5)$$

где $\gamma_{n-1} = \beta_{n-1} \beta_{n-2}$.

Тогда с учетом последних двух соотношений получим:

$$\gamma_n = \beta_n \beta_{n-1} = \frac{\|f(x_{n-1})\|}{\|f(x_n)\|} \beta_{n-1} \beta_{n-2} = \frac{\gamma_{n-1} \|f(x_{n-1})\|}{\|f(x_n)\|}. \quad (6)$$

Объединяя (5) и (6) и полагая $\gamma_0 = \beta_0^2$, получим (3). Таким образом, мы пришли к процессу (1)–(3). **Теорема доказана.**

Модифицируем формулу (3):

$$\beta_n := \min\left(1, \frac{\gamma_{n-1}}{\beta_{n-1} \|f(x_n)\|}\right), \quad \gamma_n := \frac{\gamma_{n-1} \|f(x_{n-1})\|}{\|f(x_n)\|}, \quad (7)$$

где $\gamma_0 = \beta_0^2 \|f(x_0)\|$.

Итерационный процесс, который получается из процесса (1)–(3) заменой формулы (3) на (7), обозначим как процесс (1),(2),(7).

Этот процесс также допускает альтернативное представление. Для этого введем формулу пересчета шаговой длины:

$$\beta_n = \begin{cases} \min\left(1, \frac{\|f(x_0)\|}{\|f(x_n)\|} \beta_0\right), & n = 1, \\ \min\left(1, \frac{\|f(x_{n-2})\|}{\|f(x_n)\|} \beta_{n-2}\right), & n = 2, 3, \dots \end{cases} \quad (8)$$

Итерационный процесс, который получается из процесса (1)–(3) заменой формулы (3) на формулу (8), обозначим как процесс (1),(2),(8).

Сходимость итерационного процесса (2),(3),(9) при различных ограничениях на гладкость оператора f можно доказать точно так же, как и для процесса (1)–(3). Справедлива

Теорема 2. Итерационные процессы (1),(2),(7) и (1),(2),(8) при одинаковых начальных параметрах (x_0, β_0) эквивалентны (т.е. генерируют одну и ту же последовательность приближений $\{x_n\}$).

На базе (4) и (8) построим следующие формулы пересчета шаговой длины:

$$\beta_n = \begin{cases} \min\left(1, \frac{\|f(x_0)\|}{\|f(x_n)\|} \beta_0\right), & n = 1, \\ \min\left(1, \frac{\lambda \|f(x_{n-2})\| + (1 - \lambda) \|f(x_{n-1})\|}{\|f(x_n)\|} \beta_{n-2}\right), & n = 2, 3, \dots \end{cases} \quad (9)$$

$$\beta_n = \begin{cases} \min\left(1, \frac{\|f(x_0)\|}{\|f(x_n)\|} \beta_0\right), & n = 1, \\ \min\left(1, \frac{\|f(x_{n-2})\|^\lambda \cdot \|f(x_{n-1})\|^{1-\lambda}}{\|f(x_n)\|} \beta_{n-2}\right), & n = 2, 3, \dots \end{cases} \quad (10)$$

Итерационные процессы, которые получаются из процесса (1)–(3) заменой формулы (3) на формулы (9) и (10), обозначим как процессы (1),(2),(9) и (1),(2),(10) соответственно.

Формулы (9) и (10) при надлежащем выборе параметра λ позволяют смягчить резкие колебания норм невязок каждого из процессов. В то же время необходимо задавать λ таким образом, чтобы вклад каждого из значений $\|f(x_{n-2})\|/\|f(x_n)\|$ и $\|f(x_{n-1})\|/\|f(x_n)\|$ в сумму (произведение) был пропорционален их величине. Это позволит предотвратить необоснованное замедление процесса. Очевидным выбором является следующий способ просчета λ :

$$\lambda = \frac{\|f(x_{n-2})\|}{\|f(x_{n-2})\| + \|f(x_{n-1})\|}$$

Сходимость итерационных процессов (1),(2),(9) и (1),(2),(10) при различных ограничениях на гладкость оператора f доказывается совершенно аналогично сходимости процессов (1),(2),(9) и (1),(2),(10).

В качестве тестовых задач были взяты одномерное уравнение

$$\arctg 10x = 0$$

и комбинированная система из [3]

$$\begin{cases} x_i + \sum_{j=1}^n x_j = n + 1, \quad i = \overline{1, n-3}, \\ x_1 x_2 \dots x_{n-3} x_{n-2} x_{n-1} x_n = 1, \\ \sin^2 x_1 + \cos^3 x_n = \sin^2 1 + \cos^3 1, \\ \arctg x_1 + \arctg x_n = 2 \arctg 1. \end{cases}$$

Результаты численных экспериментов говорят о том, что пересчет шаговой длины по формулам (10) и (11) позволяет ощутимо повысить процент сходимости соответствующего численного метода.

Литература

1. Мадорский, В.М. Об эффективных методах получения приближенных решений нелинейных дифференциальных задач / В.М. Мадорский, Н.Н. Стрилец // Труды Института математики. – 2004. – Т. 12, №2. – С. 130–132.
2. Стрилец, Н.Н. Об одном полностью регуляризованном нелокальном квазиньютоновском методе решения нелинейных задач / Н.Н. Стрилец // Известия Нац. академии наук Беларуси. Сер. физ.-матем. наук. – 2008. – №1. – С. 16–22.
3. Мадорский, В.М. Квазиньютоновские процессы для решения нелинейных уравнений / В.М. Мадорский. – Брест, 2005.

УДК 517.948

О ВОССТАНОВЛЕНИИ СЕТОЧНЫХ РЕШЕНИЙ НЕЛИНЕЙНЫХ ГРАНИЧНЫХ ЗАДАЧ ПОЛИНОМАМИ ЧЕБЫШЕВА

Трофимов А. В.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Мадорский В.М., кандидат физ.-мат. наук, доцент*

Конечно-разностным методом решалась задача:

$$L[x, y(x), y'(x), y''(x)] = f(x), \quad x \in [a, b],$$

$$L(y(a), y'(a)) = 0,$$

$$L(y(b), y'(b)) = 0.$$

В качестве конкретных представителей задачи рассматривались следующие три нелинейные краевые задачи:

$$y''(x) - (1 - y^2(x))y'(x) + y(x) = \frac{1}{9}(x^8 + 3x^3 - 9x^2 + 18x), \quad 0 \leq x \leq 3,$$

1. $y(0) = 0, \quad y(3) = 9,$

$$y(x) = \frac{x^3}{3}.$$

$$y''(x) + y'(x) + y(x) + y^3(x) = \cos x + \sin^3 x, \quad -\pi \leq x \leq \pi,$$

2. $y(-\pi) = y(\pi) = 0,$

$$y(x) = \sin x.$$

$$y''(x) + (y'(x))^2 + y(x) + y^2(x) = 1, \quad 0 \leq x \leq 2\pi,$$

3. $y(0) = y(2\pi) = 0,$

$$y(x) = \sin x.$$

Для восстановления в аналитическом виде приближённых сеточных решений использовалась неравномерная сетка, состоящая из n корней полинома Чебышева I рода.

Невязка на приближённые решения находилась по формуле:

$$\|L[x, y(x), y'(x), y''(x)] - f(x)\|_{L_2[a,b]}.$$

При решении задачи использовалось 3-ёх точечная аппроксимация производных по системе неравномерных узлов.

Нелинейная система решалась одним из нелокальных сверхлинейных итерационных процессов.[1]

Шаг 1. Решается линейное уравнение относительно Δx_n :

$$f'(x_n)\Delta x_n = -f(x_n), \quad n = 0, 1, 2, \dots$$

Шаг 2. Вносится поправка в вектор x_n :

$$x_{n+1} = x_n + \beta_n \Delta x_n, \quad n = 0, 1, 2, \dots, \quad \beta_0 \in [10^{-3}, 10^{-1}].$$

Шаг 3. Если $\|f(x_{n+1})\| < \varepsilon$, где ε - малая величина (параметр останова), то конец просчётов, иначе переход на шаг 4.

Шаг 4. Определяется новая шаговая длина: если $\|f(x_{n+1})\| < \|f(x_n)\|$, то $\beta_{n+1} = 1$ иначе

$$\beta_{n+1} = \min\left(1, \frac{\|f(x_n)\|\gamma_n}{\|f(x_{n+1})\|\beta_n}\right), \gamma_{n+1} = \frac{\|f(x_n)\|\gamma_n\beta_{n+1}}{\|f(x_{n+1})\|\beta_n}, n = 0, 1, 2, \dots, \gamma_0 = \beta_0^2$$

и осуществляется переход на шаг 1.

При тех же входных данных для всех трёх задач в таблице приведены средние арифметические невязок.

N запуска	Количество узлов		
	n=10	n=15	n=20
1	5,18939919947691E-17	6,29312212895089E-17	3,43767148499694E-17
2	3,56378524417662E-17	1,74632476698947E-16	1,80382886098853E-19
3	2,43007726719648E-18	2,43555942616395E-15	2,4967436032531E-18
4	2,4374222095381E-18	2,47662248315181E-15	2,55625438697336E-18

Аппроксимация полиномами Чебышева I рода весьма эффективна даже при малом количестве членов отрезка ряда Фурье.

Литература

1. Мадорский, В.М. Квазиньютоновские процессы решения нелинейных уравнений: монография / В.М. Мадорский. – Брест: Изд-во БрГУ, 2005. – 186 с.

УДК 519.95

ОПЕРАТОРНЫЙ МЕТОД РЕШЕНИЯ ЛИНЕЙНЫХ ДУ И СИСТЕМ С РАЗРЫВНОЙ ПРАВОЙ ЧАСТЬЮ

Тухто Н.Н.

*УО «Брестский государственный технический университет», г. Брест
Научный руководитель – Тузик Т.А., доцент*

В электротехнике очень часто приходится иметь дело с сигналами, которые нельзя представить в виде непрерывной функции. Так, например, в теории автоматического управления для исследования качества регулирования широко используются ступенчатые функции. Такие функции, имеющие конечное число разрывов первого рода (конечных) называются разрывными.

Уравнение, описывающее электрическую цепь, как правило является линейным дифференциальным уравнением (ЛДУ) первого или высших порядков. Поэтому возникает необходимость в решении таких уравнений с разрывной правой частью.

Будем решать уравнения с помощью преобразования Лапласа [1-2]. В данной работе рассмотрены некоторые уравнения и системы уравнений такого типа.

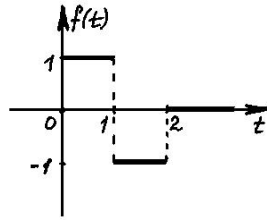
Рассмотрим следующие примеры:

1. Найти решение дифференциального уравнения $x'' + x = f(t)$

$$f(t) = \begin{cases} 1, & \text{àñë 0} \leq t < 1, \\ -1, & \text{àñë 1} \leq t < 2, \\ 0, & \text{àñë } t \geq 2. \end{cases}$$

при нулевых начальных условиях: $x(0) = 0, x'(0) = 0$.

Функцию $f(t)$ можно представить графически:



Запишем функцию $f(t)$ единым аналитическим выражением:

$$f(t) = 1\eta(t) - 1\eta(t-1) + (-1)\eta(t-1) - (-1)\eta(t-2) = 1\eta(t) - 2\eta(t-1) + 1\eta(t-2);$$

Изображение данной функции имеет вид:

$$F(p) = \frac{1}{p} - \frac{2}{p}e^{-p} + \frac{1}{p}e^{-2p} = \frac{1}{p}(1 - 2e^{-p} + e^{-2p});$$

Переходим к операторному уравнению, учитываем нулевые начальные условия и полагаем $x(t) \doteq X(p)$, $x'(t) \doteq pX(p)$, $x''(t) \doteq p^2 X(p)$.

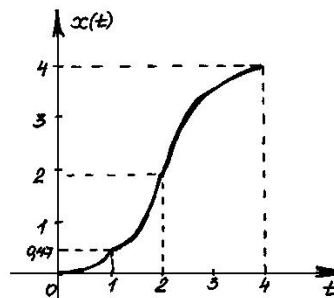
$$p^2 X(p) + X(p) = \frac{1}{p}(1 - 2e^{-p} + e^{-2p}); \quad X(p) = \frac{1}{p(p^2 + 1)}(1 - 2e^{-p} + e^{-2p});$$

т.к. $\frac{1}{p(p^2 + 1)} = \frac{1}{p} - \frac{p}{p^2 + 1} \doteq 1 - \cos t$, то применяя теорему запаздывания, находим

обычное решение данного ДУ:

$$x(t) = 1 - \cos t - 2(1 - \cos(t-1))\eta(t-1) + (1 - \cos(t-2))\eta(t-2);$$

$$x(t) = \begin{cases} 1 - \cos t, & \text{àñëë } 0 \leq t < 1, \\ 2 - \cos(t-1) - \cos t, & \text{àñëë } 1 \leq t < 2, \\ 3 - \cos(t-2) - \cos(t-1) - \cos t, & \text{àñëë } t \geq 2. \end{cases}$$



2. Найти решение системы дифференциальных уравнений:

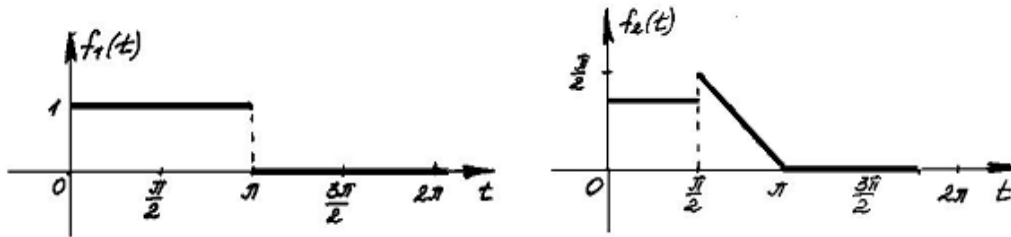
$$\begin{cases} x'' - y' = f_1(t), \\ y' + x = f_2(t). \end{cases}$$

$$f_1(t) = \begin{cases} 1, & \text{àñëë } 0 \leq t < \pi, \\ 0, & \text{àñëë } t \geq \pi. \end{cases}$$

$$f_2(t) = \begin{cases} t, & \text{àñëë } 0 \leq t < \frac{\pi}{2}, \\ \pi - t, & \text{àñëë } \frac{\pi}{2} \leq t < \pi, \\ 0, & \text{àñëë } t \geq \pi. \end{cases}$$

при нулевых начальных условиях: $x(0) = y(0) = 0$, $x'(0) = y'(0) = 0$.

Функции $f_1(t)$ и $f_2(t)$ можно представить графически:



Запишем $f_1(t)$ и $f_2(t)$ с помощью единичной функции Хевисайда:

$$f_1(t) = 1\eta(t) - 1\eta(t - \pi);$$

$$f_2(t) = t\eta(t) - t\eta(t - \frac{\pi}{2}) + (\pi - t)\eta(t - \frac{\pi}{2}) - (\pi - t)\eta(t - \pi) = t\eta(t) + (-2t + \pi)\eta(t - \frac{\pi}{2}) - (\pi - t)\eta(\pi - t);$$

Изображениями данных функций являются следующие функции:

$$F_1(p) = \frac{1}{p} - \frac{1}{p}e^{-\pi p} = \frac{1}{p}(1 - e^{-\pi p});$$

$$F_2(p) = \frac{1}{p^2} - 2\frac{1}{p^2}e^{-\frac{\pi}{2}p} + \frac{1}{p^2}e^{-\pi p} = \frac{1}{p^2}(1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p});$$

С учетом нулевых начальных условий $x(0) = y(0) = 0$, $x'(0) = y'(0) = 0$ полагаем:

$$x(t) \doteq X(p), \quad x'(t) \doteq pX(p), \quad x''(t) \doteq p^2X(p), \quad y(t) \doteq Y(p), \quad y'(t) \doteq pY(p), \quad y''(t) \doteq p^2Y(p).$$

Переходим к линейной системе относительно изображений:

$$\begin{cases} p^2X(p) - pY(p) = \frac{1}{p}(1 - e^{-\pi p}), \\ pY(p) + X(p) = \frac{1}{p^2}(1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p}). \end{cases}$$

Методом определителей решим эту систему:

$$\Delta = \begin{vmatrix} p^2 & -p \\ 1 & p \end{vmatrix} = p^3 + p = p(p^2 + 1);$$

$$\Delta_x = \begin{vmatrix} \frac{1}{p}(1 - e^{-\pi p}) & -p \\ \frac{1}{p^2}(1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p}) & p \end{vmatrix} = (1 - e^{-\pi p}) + \frac{1}{p}(1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p});$$

$$\Delta_y = \begin{vmatrix} p^2 & \frac{1}{p}(1 - e^{-\pi p}) \\ 1 & \frac{1}{p^2}(1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p}) \end{vmatrix} = (1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p}) - \frac{1}{p}(1 - e^{-\pi p});$$

Получаем операторное решение данной системы ДУ:

$$\begin{cases} X(p) = \frac{1 - e^{-\pi p}}{p(p^2 + 1)} + \frac{1}{p^2(p^2 + 1)}(1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p}), \\ Y(p) = \frac{1 - 2e^{-\frac{\pi}{2}p} + e^{-\pi p}}{p(p^2 + 1)} - \frac{1 - e^{-\pi p}}{p^2(p^2 + 1)}. \end{cases}$$

Применяя теорему запаздывания, находим решение системы ДУ:

$$\begin{cases} x(t) = 1 - \cos t - (1 - \cos(t - \pi))\eta(t - \pi) + t - \sin t - 2\left(\left(t - \frac{\pi}{2}\right) - \sin\left(t - \frac{\pi}{2}\right)\right)\eta\left(t - \frac{\pi}{2}\right) + \\ \quad + \left(\left(t - \pi\right) - \sin(t - \pi)\right)\eta(t - \pi), \\ y(t) = 1 - \cos t - t + \sin t - 2\left(\left(t - \frac{\pi}{2}\right) - \cos\left(t - \frac{\pi}{2}\right)\right)\eta\left(t - \frac{\pi}{2}\right) + (1 - \cos(t - \pi))\eta(t - \pi) + \\ \quad + \left(\left(t - \pi\right) - \sin(t - \pi)\right)\eta(t - \pi). \end{cases}$$

Литература

1. Краснов, М.Л. Функции комплексного переменного. Операционное исчисление. Теория устойчивости / М.Л. Краснов, А.И. Киселев, Г.И., Макаренко – М.: Наука, 1981.
2. Сборник задач по математике для втузов специальные разделы математического анализа – М.: Наука, 1981

УДК 519.216.74

МОДЕЛИРОВАНИЕ ВЛИЯНИЯ УСТОЙЧИВОГО СТОХАСТИЧЕСКОГО ВОЗМУЩЕНИЯ НА СИСТЕМУ УРАВНЕНИЙ РЁССЛЕРА

Черноокий А.Л.

*УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель Труш Н.Н – доктор физико-математических наук, профессор*

В динамике химических реакций с перемешиванием известна довольно простая модель Рёслера [1], задаваемая системой обыкновенных дифференциальных уравнений:

$$\begin{aligned} \frac{dx}{dt} &= -y - z, \\ \frac{dy}{dt} &= x + ey, \\ \frac{dz}{dt} &= f + xz - mz, \end{aligned} \tag{1}$$

где x, y, z – компоненты решения, e, f, m – параметры модели.

Будем рассматривать модель (1) со стохастическими возмущениями в виде системы стохастических дифференциальных уравнений с устойчивыми приращениями:

$$\begin{aligned} dx &= (-y - z)dt, \\ dy &= (x + ey), \\ dz &= (f + xz - mz)dt + cdL_{\alpha,\beta}(t), \end{aligned} \quad (2)$$

где c – параметр возмущения, $dL_{\alpha,\beta}(t)$ – β -устойчивый процесс Леви, имеющий фрактальную структуру и неограниченную дисперсию. Алгоритм моделирования процесса $dL_{\alpha,\beta}(t)$ можно взять из [2].

Для моделирования системы (2) на отрезке $[0; T]$ воспользуемся неявным методом Эйлера-Маруаи для стохастических дифференциальных уравнений с устойчивыми приращениями[3]. Для этого разобьем отрезок интегрирования на N равных отрезков точками $t_i = i \frac{T}{N}, i = \overline{0, N}$ с шагом $h = \frac{T}{N}$, при заданных начальных условиях x_0, y_0, z_0 . В результате получим следующие выражения для дискретного численного метода, моделирующего значения траектории уравнения (2) в узлах сетки:

$$\begin{aligned} x_{i+1} &= x_i + h(-y_{i+1} - z_{i+1}), \\ y_{i+1} &= y_i + h(x_{i+1} + ey_{i+1}), \\ z_{i+1} &= z_i + h(f + x_{i+1}z_{i+1} - mz_{i+1})dt + c(L_{\alpha,\beta}(t_{i+1}) - L_{\alpha,\beta}(t_i)). \end{aligned} \quad (3)$$

где $L_{\alpha,\beta}(t_{i+1}) - L_{\alpha,\beta}(t_i)$ – независимые приращения процесса Леви, имеющие устойчивое распределение.

Для решения системы нелинейных уравнений (3) воспользуемся методом Ньютона.

Возьмем $e=0.19, f=0.4, m=8.5, x_0=15, y_0=10, z_0=10$. Результат моделирования траектории системы (1) на фазовой плоскости (x, y) представлен на рисунке 1.

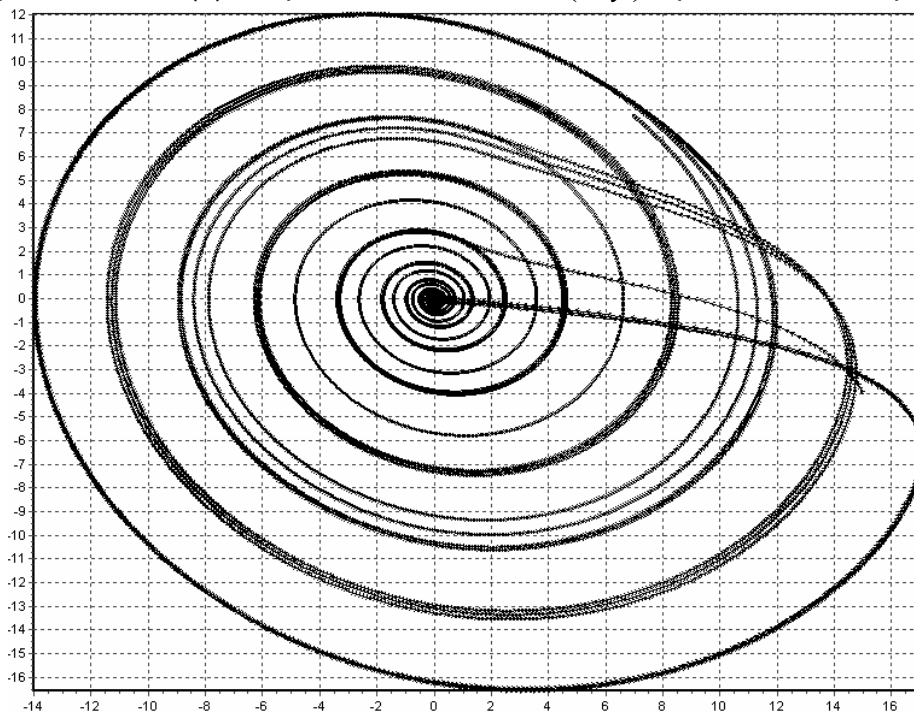


Рисунок 1 – «Слоистый» аттрактор Рёслера без возмущений

Аттрактор Рёслера возникает как результат последовательных бифуркаций удвоения периода, которые начинаются при $m=3.5$ и приводят при $m=4.2$ к появлению ат-

трактора [4]. Это приводит к появлению «полос» с наиболее плотным прохождением траекторий, сам Рёсслер назвал этот аттрактор «слоистым».

Теперь будем последовательно увеличивать параметр возмущения, пока траектория не выйдет из области притяжения аттрактора. Траектория системы (2) на фазовой плоскости (x, y) при $c = 3.5$ представлена на рисунке 2.

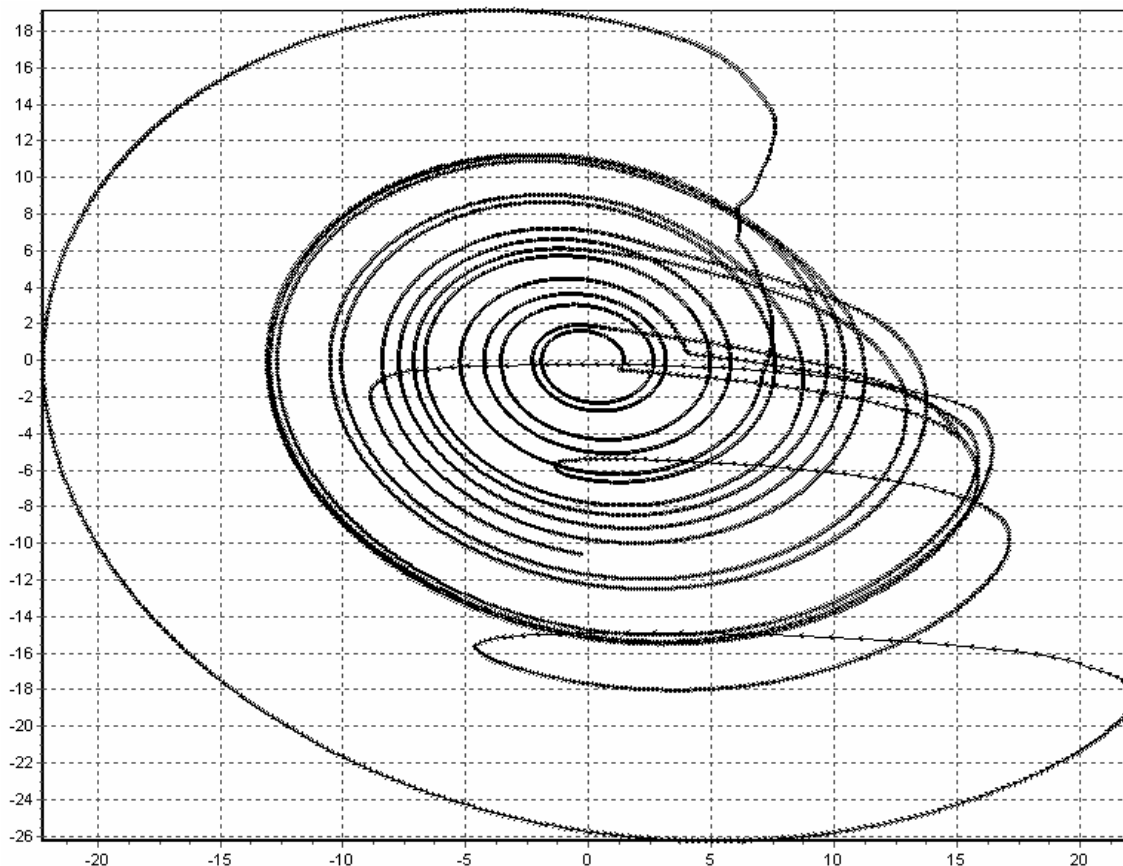


Рисунок 2 – Траектория системы Рёсслера в плоскости (x, y) со стохастическими устойчивыми возмущениями при $c=3.5$

Как видно из рисунка, «слоистость» аттрактора сохраняется, что свидетельствует об устойчивости неявного метода Эйлера-Маруями.

Литература

1. Ressler, O.E. "Chemical Turbulence: Chaos in a Small Reaction-Diffusion System", Z. Naturforsch. a 31,1168-1172.
2. Janicki, A., Izydorczyk A. Komputerowe metody w modelowaniu stochastycznym. Warszawa. Wydawnictwa Naukowe-Tehniczne. 2001.
3. Труш, Н.Н., Черноокый, А.Л. Численное моделирование стохастических дифференциальных уравнений с устойчивыми возмущениями. // Известия Нац. академии наук Беларуси. Сер. физ.-матем. наук, 2009. – No.2 – С. 5–10.
4. Неймарк Ю.И., Ланда, П.С. — Стохастические и хаотические колебания. – М.: Наука, 1987. – 422 с.

УДК 517.925

ПОДВИЖНЫЕ ОСОБЫЕ ТОЧКИ СИСТЕМЫ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Чуприна О.В.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
Научный руководитель – Мельникова И.Н., кандидат физ.-мат. наук, доцент

В работе рассматривается система дифференциальных уравнений вида

$$\frac{dx_i}{dz} = f_i(x_1, x_2, \dots, x_n, z) \quad (i=\overline{1, n}), \quad (1)$$

где f_i – однозначные аналитические функции по переменным x_1, x_2, \dots, x_n, z в области $|x_i - x_i^0| < p$, $|z - z_0| < r$, т.е. в окрестности точки $M_0(x_1^0, x_2^0, \dots, x_n^0, z)$, где x_i^0 ($i = \overline{1, n}$) и z_0 – конечные комплексные числа.

Наряду с задачей Коши в аналитической теории дифференциальных уравнений весьма важной является задача изучения решений во всей области их существования. Но поведение аналитической функции и область ее существования, как известно, определяется ее особыми точками. Наибольший интерес представляет исследование вопроса о характере подвижных особых точек, положение которых на плоскости зависит от начальных условий, определяющих решение системы дифференциальных уравнений. Знание ответов на вопрос, могут ли системы уравнений вида (1) иметь решения с подвижными особыми линиями, а также с отдельными особыми точками, очень важно, т.к. без них мы не можем дать общую характеристику решений того или другого класса указанных систем. В работе сделана попытка обобщить результаты статей [1], [2] на случай системы (1).

В работе используется метод исследований, разработанный С.Г. Кондратеней, в основу которого положено разделение существенно особых подвижных точек аналитической функции на два класса и обобщение теоремы единственности решения системы (1) на случай особых начальных условий.

Литература

1. Кондратеня, С.Г. Характеристика сложности особых точек решений нелинейных систем двух дифференциальных уравнений / С.Г. Кондратеня // Диф. уравнения. – 1979. – Том 15., № 9. – С. 1580–1591.
2. Кондратеня, С.Г. Классы систем трех дифференциальных уравнений, не имеющих решений с неопределенными компонентами / С.Г. Кондратеня, И.Н. Мельникова // Диф. уравнения. – 1993. – Том 29., № 6. – С. 1069–1070.

ПОСТРОЕНИЕ СИСТЕМЫ ЭКВИВАЛЕНТНОЙ ДИФФЕРЕНЦИАЛЬНОМУ УРАВНЕНИЮ ШАЗИ С ШЕСТЬЮ ОСОБЫМИ ТОЧКАМИ

Швычкина Е. Н.

УО «Брестский государственный технический университет», г. Брест

Уравнения третьего порядка с шестью различными полюсами относительно искомой функции рассматривал Шази [4]. Результаты его исследований сводятся к следующему.

1) Если решения таких уравнений не имеют подвижных критических особых точек, то последние по необходимости должны иметь вид

$$w''' = \sum_{k=1}^6 \frac{(w'-a_k')(w''-a_k'') + A_k(w'-a_k')^3 + B_k(w'-a_k')^2}{w-a_k} + \frac{C_k(w'-a_k')}{w-a_k} + Dw'' + Ew' + \prod_{i=1}^6 (w-a_i) \sum_{k=1}^6 \frac{F_k}{w-a_k}, \quad (1)$$

которые, очевидно, содержит 32 функции по z : A_k, B_k, C_k, F_k ($k = \overline{1,6}$), и D, E .

2) Если, кроме того, эти функции удовлетворяют специальной системе (S), состоящей из 31 алгебраического и дифференциального уравнения, то эти условия будут и достаточными. Система (S) содержит 31 уравнение с 32 функциями.

Однако Шази не завершил интегрирование системы (S) и не определил класс неприводимых уравнений, которому принадлежало бы уравнение (1). Вместе с тем он показал, что некоторые случаи вырождения уравнения (1) являются неприводимыми уравнениями Пенлеве. С этой точки зрения, уравнение (1) можно рассматривать как новое, трансцендентные решения которого не имеют подвижных критических особых точек [1].

При исследовании характера особых точек во многих случаях удобнее рассматривать вместо уравнения (1) эквивалентную ему систему.

В данной работе, для случая, когда коэффициенты уравнения (1) являются постоянными и $B_k = 0$ ($k = \overline{1,6}$), составлена система вида

$$\begin{cases} w'' = -f_1(z, w)w'^2 + f_2(z, w)v + f_3(z, w), \\ v' = -2f_1(z, w)w'v + f_4(z, w); \end{cases} \quad (2)$$

где f_i ($i = \overline{1,4}$) функции по z и w подлежащие определению. Продифференцируем первое уравнение системы (2) по z и, подставляя (2) в уравнение Шази (1) и сравнивая коэффициенты при одинаковых степенях w , получим систему дифференциальных уравнений для отыскания функции f_i ($i = \overline{1,4}$). Эта система имеет следующий вид

$$\begin{aligned} f_2 \sum_{k=1}^6 \frac{1}{w-a_k} - \frac{\partial f_2}{\partial w} = 0, \quad f_2 - \frac{\partial f_2}{\partial z} = 0, \quad f_1 - \frac{\partial f_1}{\partial z} = 0, \\ -2f_1^2 + f_1 \sum_{k=1}^6 \frac{1}{w-a_k} + \sum_{k=1}^6 \frac{A_k}{w-a_k} - \frac{\partial f_1}{\partial w} = 0, \end{aligned} \quad (3)$$

$$E - 2f_3f_1 + f_3 \sum_{k=1}^6 \frac{1}{w - a_k} + \sum_{k=1}^6 \frac{C_k}{w - a_k} + \frac{\partial f_3}{\partial w} = 0,$$

$$f_4f_2 + \frac{\partial f_3}{\partial z} = 0.$$

Из первых двух уравнений системы (3) находим функцию

$$f_2(z, w) = z \cdot \prod_{k=1}^6 (w - a_k) \cdot C_1,$$

где C_1 – произвольная постоянная. Чтобы решить систему (3) применим метод, который рассматривается в работе [3] для решения линейного уравнения второго порядка класса Фукса с шестью особыми точками. Рассмотрим подробнее процедуру нахождения функции $f_1(z, w)$. Из третьего уравнения системы (3) найдем

$$f_1(z, w) = z \cdot T(w).$$

Т. о. четвертое уравнение системы (3) является уравнением Риккати относительно неизвестной функции $T(w)$. Решение будем искать в виде

$$T(w) = \frac{b_0w^5 + b_1w^4 + b_2w^3 + b_3w^2 + b_4w + b_5}{w^6 - w^5\sigma_1 + w^4\sigma_2 - w^3\sigma_3 + w^2\sigma_4 - w\sigma_5 + \sigma_6}, \quad (4)$$

где σ_i ($i = \overline{1,6}$) – элементарные симметрические многочлены, составленные из элементов a_i ($i = \overline{1,6}$). Преобразуем следующее выражение к виду

$$\sum_{k=1}^6 \frac{A_k}{w - a_k} = \frac{6w^4 - 3w^2\alpha + 3w\beta_2 - 3\beta_3 - 4w^3\sigma_1 + 3w^2\sigma_2 + \sigma_4}{w^6 - w^5\sigma_1 + w^4\sigma_2 - w^3\sigma_3 + w^2\sigma_4 - w\sigma_5 + \sigma_6}, \quad (5)$$

где σ_i ($i = \overline{1,6}$) связаны с величинами α , β_2 , β_3 следующими соотношениями,

$$\beta_2 = \frac{(3\alpha_2 - \sigma_2)(2\alpha_2\sigma_1 - 3\sigma_3) - 2\sigma_1\sigma_4 + 6\sigma_5}{18\alpha_2 + \sigma_1^2 - 6\sigma_2},$$

$$\beta_3 = \frac{2\sigma_1\sigma_5 - (3\alpha_2 - \sigma_2)(12\alpha_2^2 - 4\alpha_2\sigma_2 + \sigma_1\sigma_3 - 4\sigma_4) - 2\sigma_1\sigma_4 + 6\sigma_5}{2(18\alpha_2 + \sigma_1^2 - 6\sigma_2)},$$

а число α удовлетворяет уравнению 5-й степени [3]. Выражение (5) получено также с учетом выполнения соотношений между функциями A_i и a_i системы (S).

Подставляя (4) в четвертое уравнение системы (3), получим уравнение десятой степени относительно w . Приравнявая коэффициенты этого уравнения нулю, получим систему из одиннадцати алгебраических уравнений, где неизвестными являются коэф-

коэффициенты b_k ($k = \overline{0, 5}$) и σ_i ($i = \overline{1, 6}$). Частное решение четвертого уравнения системы (3) в случае $b_0 = 2$ есть $T(w) = \frac{2}{w}$ т.о.

$$f_1(z, w) = z \cdot \frac{2w^2(2C_2w - 3)}{2w^3(C_2w - 2) + \sigma_3}$$

Построение систем (2) и (3), нахождение их решений требует громоздких вычислений, которые были реализованы в системе *Mathematica* [2]. В данной работе не приводится явный вид всех функций f_i ($i = \overline{1, 4}$) по причине их большого объема.

Литература

1. Добровольский, В.А. очерки развития аналитической теории дифференциальных уравнений / В.А. Добровольский – Киев: Вища школа, 1974. – 456 с.
2. Чичурин, А.В. Уравнение Шази и линейные уравнения класса Фукса: Монография / А.В. Чурин – М.: Изд-во РУДН, 2003. – 143 с.
3. Chazy J. – Sur les equations differentielles du troisieme order et d'ordre superieur, don't l'integrale generale a ses points critiques fixes. Acta Math., 1911 – N 34. – P. 317 – 385.

УДК 517.983+519.6

О СХОДИМОСТИ В ГИЛЬБЕРТОВОМ ПРОСТРАНСТВЕ ИТЕРАЦИОННОГО МЕТОДА РЕШЕНИЯ ОПЕРАТОРНЫХ УРАВНЕНИЙ

Юрко И.В.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест
 Научный руководитель – Савчук В.Ф., кандидат физ.-мат. наук, доцент

Поскольку некорректные задачи возникают в многочисленных приложениях математики, то проблема их решения и разработка методов их решения является актуальной. В работе предлагается новый метод решения некорректных задач. Цель исследования – доказать сходимость предложенного метода в случае априорного выбора числа итераций, получить оценки погрешности и их минимизировать. Методологической основой исследования является общая теория некорректных задач, элементы функционального и математического анализа, вычислительной математики.

В гильбертовом пространстве H решается уравнение 1-го рода

$$Ax = y, \tag{1}$$

где A - ограниченный положительный и самосопряженный оператор, в предположении, что нуль принадлежит спектру оператора A , но не является его собственным значением. Тогда задача о разрешимости уравнения (1) является некорректной. Если решение уравнения (1) существует, то будем искать его с помощью итерационного метода

$$x_{n+1} = (E - \alpha A)^2 x_n + 2\alpha y - \alpha^2 Ay, x_0 = 0. \tag{2}$$

Обычно правая часть уравнения (1) известна с некоторой точностью δ , т.е. известно y_δ такое, что $\|y - y_\delta\| \leq \delta$. Тогда метод (2) примет вид

$$x_{n+1,\delta} = (E - \alpha A)^2 x_{n,\delta} + 2\alpha y_\delta - \alpha^2 A y_\delta, x_{0,\delta} = 0. \quad (3)$$

Изучена сходимость метода (2) при точной правой части y уравнения (1). Справедлива

Теорема 1 Итерационный процесс (2) при условии $0 < \alpha < \frac{2}{\|A\|}$ сходится.

Показано, что метод (3) сходится при условии $0 < \alpha < \frac{2}{\|A\|}$, если число итераций n разумно согласовывать с уровнем погрешности δ , т.е. доказана

Теорема 2. При условии $0 < \alpha < \frac{2}{\|A\|}$ итерационный процесс (3) сходится, если выбрать число итераций n в зависимости от δ так, чтобы $n\delta \rightarrow 0, n \rightarrow \infty, \delta \rightarrow 0$.

В предположении, что точное решение уравнения (1) истокообразнопредставимо, т.е. $x = A^s z, s > 0$ получена оценка погрешности для метода (3), т.е. справедлива

Теорема 3. Если точное решение x уравнения (1) истокообразно представимо, то при условии $0 < \alpha \leq \frac{5}{4\|A\|}$ для метода (3) справедлива оценка погрешности

$$\|x - x_{n,\delta}\| \leq s^s (2n\alpha e)^{-s} \|z\| + 2n\alpha\delta. \quad (4)$$

Оптимизируем полученную оценку (4) по n , т.е. найдём такое значение числа итераций n_{opt} , при котором оценка (4) становится минимальной. Для этого производную от правой части неравенства (4) приравняем нулю. Итак, доказана

Теорема 4. Если точное решение x уравнения (1) истокообразно представимо, то при условии $0 < \alpha \leq \frac{5}{4\|A\|}$ оптимальная оценка погрешности для итерационного метода (3) имеет

$$\text{вид } \|x - x_{n,\delta}\|_{opt} \leq (1+s)e^{-\frac{s}{s+1}} \delta^{\frac{s}{s+1}} \|z\|_{\frac{1}{s+1}} \text{ и достигается при } n_{opt} = s(2\alpha)^{-1} e^{-\frac{s}{s+1}} \delta^{-\frac{1}{s+1}} \|z\|_{\frac{1}{s+1}}.$$

Метод (2) пригоден и тогда, когда $\lambda = 0$ является собственным значением оператора A (случай неединственного решения уравнения (1)). Обозначим через $N(A) = \{x \in H \mid Ax = 0\}$, $M(A)$ – ортогональное дополнение ядра $N(A)$ до H . Пусть $P(A)x$ – проекция $x \in H$ на $N(A)$, а $\Pi(A)x$ – проекция $x \in H$ на $M(A)$. Справедлива

Теорема 5. Пусть $A \geq 0, y \in H, 0 < \alpha < \frac{2}{\|A\|}$, тогда для итерационного процесса (2)

верны следующие утверждения:

а) $Ax_n \rightarrow \dot{I}(A)y, \|Ax_n - y\| \rightarrow I(A, y) = \inf_{x \in H} \|Ax - y\|;$

б) метод (2) сходится тогда и только тогда, когда уравнение $Ax = \Pi(A)y$ разрешимо. В последнем случае $x_n \rightarrow P(A)x_0 + x^*$, где x^* – минимальное решение уравнения.

СОДЕРЖАНИЕ

СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В НАУЧНЫХ И ПРИКЛАДНЫХ ИССЛЕДОВАНИЯХ	3
A. Glebov, D. Strykeleu Experimental convergence study of a genetic algorithm using consistency-preserving genetic operators for tree-like chromosomes	3
Брикет Я.С. Современные технологии WEB-программирование: CMS Joomla	5
Брич А.Л., Мухоровский С.В. Компьютерная генерация трехсвязных однородных плоских графов, не имеющих гамильтонова контура	7
Брич А.Л. Информационные технологии в обучении программированию	8
Ванюков С.В. Об организации системы тестирования	10
Герашенко Е.В. Программа XpertRule Knowledge Builder как инструмент построения деревьев решений	11
Герман Ю.В. Применение системы компьютерной алгебры Mathematica на уроках математики	14
Грисевич Л.Н. Программно-аппаратный комплекс автоматизации акустического спектрального анализа	16
Давыденко Е.А. ХАМРР – перспектива применения	19
Дёмин В.В. Реализация модели динамических мини-окон в аппаратно-ускоренном графическом интерфейсе	21
Дядищев А.О., Грязных П.А. Внутренняя архитектура системы тестирования e-Judge	25
Емельянов Д.С, Трохова Т. А. Компьютерное моделирование процесса центрирования обсадных колонн для нефтяных скважин	26
Жешко Н.С., Тарариев А.И. Компьютерное тестирование	28
Игнатенко В.Ю. Модель расчета экономического раздела ДП	29
Копылов Д.А., Никонюк А.Н. Информационно-поисковая система “Научный потенциал вуза”	32
Корольчук Д.В. MOODLE как основа организации системы дистанционного обучения математике	35
Кочурко В.А. Сравнительный обзор «быстрого» алгоритма генерации графов	38
Крагель Е.А. Особенности использования информационных технологий для адаптации уровня математической подготовки студентов-иностранцев	40
Логвинович В.Я., Ходин В.М. Проект Информационной среды Управления вузом на основе системы 1С: Предприятие	42
Мацкевич С.В., Савицкий Ю.В. Интерактивные анимационные программные комплексы в системе технического образования	44
Мельничук А.В, Полуянович Л.В., Савицкий Ю.В. Проблемы проектирования структурированных кабельных систем локальных вычислительных сетей	47
Морозенко Г.П., Сидорович А.С. Применение бескликовой навигации в графическом интерфейсе пользователя	48
Мулькевич Н.Н., Хвещук В.И. Оценка ИТ-проектов	52
Ничипорук Т.С. Автоматизированные банковские информационные системы ЗАО «Софтклуб»	55
Охримчук В.В. Математические модели защиты информационных ресурсов малых предприятий	58

Райчёнок С.В. Некоторые способы рисования графов на плоскости	60
Рыжков А.А. Тестирование задач олимпиадного характера	63
Семёнов Г.А. Использование пакета “Signal Express” для регистрации кинетики генерации лазера	65
Сергиенко Е.Ю. О методах анализа неоднородных временных рядов гидрометеорологической информации	67
Сечко Ю.Н., Матюшков Л.П. Анализ подходов к решению NP-полных задач	69
Склипус Д.Б. Автономный мобильный микроконтроллерный стенд	72
Согоян А. Л. Древовидный алгоритм разбиения слов русского языка на слоги на основе статистических данных	76
Тавониус К.А. Использование периферического зрения при навигации между страницами в Интернет-браузере	77
Тараканова О.М. Динамическое определение объема совершаемой сделки как способ оптимизации механической торговой системы	80
Теут А.А. Общая архитектура классов доступа к базам данных	82
Титова Т.И. AllFusion Process Modeler 7 (BPwin) как средство функционального моделирования бизнес-процессов	83
Ткач С.Н. Общие требования подготовки электронных презентаций	85
Ткачёнок Е.М. Метод анализа иерархий в управленческих задачах	88
Тухто Н.Н. Применение диагональной прокрутки в графическом интерфейсе мобильных устройств	92
Фарберов А. Г., Трохова Т. А. Автоматизация проектирования гидродинамики цементирования нефтяных скважин	95
Хинич А.И. Моделирование уровней подземных вод при разработке торфяных месторождений (на примере “Гричино-Старобинского” торфяного месторождения)	98
Шуть В.Н., Войцехович О.Ю., Анфилец С.В. Оптимизация и координация управления светофорами	102

АНАЛИТИЧЕСКИЕ И ЧИСЛЕННЫЕ МЕТОДЫ ИССЛЕДОВАНИЙ В МАТЕМАТИКЕ И ИХ ПРИЛОЖЕНИЯ

Альциванович Л.М. Сходимость метода итераций явного типа в энергетической норме гильбертова пространства	105
Артюшеня А.А. Об эффективности одношаговых и многошаговых методов решения нелинейных систем	107
Белемук О.В. О построении общего интеграла у нелинейных обыкновенных дифференциальных уравнений 4-го порядка	110
Берёзкина М.С. Сходимость одного метода решения линейных уравнений с неограниченным оператором в гильбертовом пространстве	112
Божко И.Н., Дацык В.Т. Асимптотические формулы типа Вороновской	113
Будько Д. А. Нормализация систем линейных гамильтоновых уравнений	116
Велесевич А.И., Дацык В.Т. Представление некоторых функций двух переменных сингулярными интегралами в точках пересечения их линий разрыва	119
Гетман В.А., Гладкий И.И., Ефимик С.В., Махнист Л.П. Сходимость и свойства решения одного стохастического уравнения	121
Грицук Д.В., Климашевская И.Н. Об одном классе систем дифференциальных уравнений	127

Дереченник С.С. Ограничения метода наименьших квадратов при построении моделей линейной регрессии	128
Дыбко А.В., Дацык В.Т. Аппроксимационные методы вычисления интегралов	131
Казымов Б.З., Насирова К.К. Численная расчетная модель для определения основных характеристик разработки газовых залежей с ползучей средой	133
Калач А.В., Шепшелей Д.В. О восстановлении сеточного решения линейной дифференциальной задачи второго порядка	134
Кожух И.Г., Корделюк О.И. Математическая модель одного биологического процесса	136
Кожух И.Г., Касперович Ю.А. Метод медленно меняющихся амплитуд в исследовании динамических систем	139
Коробейникова Е.В. Стационарное распределение сети с групповыми перемещениями в обобщенной мультипликативной форме	142
Костюк А.Ю. Метод секущих для минимизации функций одной переменной	146
Кулак Т.Н. Исследование математического ожидания и дисперсии осредненной оценки взаимной спектральной плотности	149
Лапич С.В. Интеграл Дюамеля и его приложения	152
Лялюк Ю.В. Решение операторных уравнений двухшаговым итерационным методом	155
Марчук А.Ю. Исследование математического ожидания состоятельной оценки взаимной спектральной плотности	156
Наумовец С.Н. Останов по соседним приближениям в методе итераций явного типа решения некорректных задач	159
Наумовец С.Н. К вопросу о сходимости явного итерационного метода решения некорректных задач	160
Онищук А.И. Аналог задачи Коши для нелинейного дифференциального уравнения со старшей производной дробного порядка	162
Охримчук А.В., Дацык В.Т. Об одном классе нелинейных дифференциальных уравнений со старшей производной нецелого порядка	163
Савчук Л.А. О фундаментальном решении оператора Δ^2 в R^3	164
Степанюк Г.П., Чичурин А.В. К вопросу интегрируемости в квадратурах линейного дифференциального уравнения второго порядка с шестью особыми точками	167
Стрилец Н.Н. О двухпараметрических квазиньютоновских итерационных методах для решения нелинейных уравнений	169
Трофимов А. В. О восстановлении сеточных решений нелинейных граничных задач полиномами Чебышева	173
Тухто Н.Н. Операторный метод решения линейных ДУ и систем с разрывной правой частью	174
Черноокий А.Л. Моделирование влияния устойчивого стохастического возмущения на систему уравнений Рёсслера	177
Чуприна О.В. Подвижные особые точки системы дифференциальных уравнений	180
Швычкина Е.Н. Построение системы эквивалентной дифференциальному уравнению Шази с шестью особыми точками	181
Юрко И.В. О сходимости в гильбертовом пространстве итерационного метода решения операторных уравнений	183

Научное издание

СОВРЕМЕННЫЕ ПРОБЛЕМЫ МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Материалы VI Республиканской научной конференции
молодых ученых и студентов
26–28 ноября 2009 года

Часть II

Текст печатается в авторской редакции

Ответственный за выпуск: **Махнист Л.П.**

Редактор: **Строкач Т.В.**

Компьютерная вёрстка: **Боровикова Е.А.**

Корректор: **Никитчик Е.В.**

Лицензия № 02330/0549435 от 8.04.2009 г.

Подписано к печати 11.11.2009 г.

Формат 60x84 ¹/₁₆. Бумага «Снегурочка».

Гарнитура Arial Narrow. Усл. печ. л. 11,0. Уч.-изд. л. 11,75.

Заказ N 1015. Тираж 112 экз.

Отпечатано на ризографе учреждения образования
«Брестский государственный технический университет».

224017, г. Брест, ул. Московская, 267.